UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

MASTER THESIS

Development of Models for the Equations of Motion in the Solar System: Implementations and Applications

Anshuk Attri

SUPERVISED BY

Prof. Josep J. Masdemont

Universitat Politècnica de Catalunya
Master in Aerospace Science & Technology
July 2014

This Page Intentionally Left Blank

Development of Models for the Equations of Motion in the Solar System: Implementations and Applications

BY

Anshuk Attri

DIPLOMA THESIS FOR DEGREE

Master in Aerospace Science and Technology

AT

Universitat Politècnica de Catalunya

SUPERVISED BY:

Prof. Josep J. Masdemont

Departament de Matemàtica Aplicada I

This Page Intentionally Left Blank

ABSTRACT

There are two primary objectives of this master's thesis. The first is to develop suitable semi-analytical quasi-periodic models for the equations of motion in the solar system. The results of the research have to be implemented developing specific vector fields for the propagation of trajectories and the corresponding changes of coordinates. The second objective of the thesis is to implement a program that is able to read ephemeris data and is capable of performing numerical calculation in vector fields. After the program has been implement is has to be tested to ensure robustness. The software package has been implemented in FORTRAN and has the ability to read and access the SpiceLib of JPL, CIT, to get ephemeris data for states of bodies in the solar system. The methodology of the thesis involves: development of vector fields for restricted n-body problems, general explanations of Lagrangian and Hamiltonian forms of the restricted n-body problem, development of semi-analytical model of n-body problem, development of variational equations for n-body problems in the solar system and for quasi-periodic formulation of equations of restricted n-body problem, implementation of all these formulations in form of a package, which is able to read and access ephemeris data using SpiceLib from JPL, written in FORTRAN 77, testing the routines, and finally, performing trajectory refinement of two Libration point orbits: one in Jupiter-Europa System and one in Saturn-Titan System using Parallel Shooting.

This Page Intentionally Left Blank

Table of Contents

This Page Intentionally Left Blank

List of Figures

This Page Intentionally Left Blank

List of Tables

This Page Intentionally Left Blank

INTRODUCTION

Study of orbital dynamics is very important for space mission design. It is interesting both from a mathematical as well as engineering point of view. Orbital dynamics is useful for computing trajectories of artificial satellites, asteroids, comets etc. The trajectory of a body moving in a gravitational field can be represented in wide varieties of models. The ease of integrations in a model depends on its simplicity. The most basic model is using Newton's laws of gravitation and to perform integrations based on this model.

Models of motion in a gravitational field can be entirely analytical i.e. defined in terms of basic equations of motion. The problem with this model is that numerical computations can take a significant amount of time. Thus, sometimes, these model can be refined and we can move them in the domain of semi-analytical. This allows to perform integrations in the numerical domain, reducing the complexity of the problem. The aim of the having semi-analytical models is the possibility to have intermediate models (approximations) between simple models like RTBP and full ephemeris models (either numerical or analytical).

There are two primary aims of this master's thesis. The first is to develop suitable semi-analytical quasi-periodic models for the equations of motion in the solar system. These models will be implemented through vector fields and the trajectories will be propagated in these vector fields. The second objective of the thesis is to implement a program that is able to read ephemeris data.

The package that is being designed has the capability to read the ephemeris by Jet Propulsion Laboratory (JPL), CIT. The package is also be capable of performing numerical calculation in vector fields. The work on the package was started as a part of Reference [7]. Before the work on this thesis was started the package had the ability to read the ephemeris and to perform change of coordinates. As the work progressed, it was discovered that the package had some errors and the features were not working as they should. These bugs were removed before the new vector fields were implemented. After the program routines were implemented it was tested to ensure robustness and accuracy. Finally, the the package was used to perform trajectory refinement of two Libration point orbits: one in Jupiter-Europa System and one in Saturn-Titan System using Parallel Shooting.

The first chapter of this thesis will formulate the equations of motion of a mass less body in the solar system. Also, the restricted three body problem will be elaborated and the equations required for the computations will be derived. It will also introduce the concept of reference frames and the advantages of transforming between them. It will also introduce the Lagrangian, Hamiltonian, and the Quasi-Periodic formulation of the restricted n-body problem.

The second chapter introduces the concept and the mathematical background of variational equations. The variational equations will then be formulated for n-body problem with respect to the solar system barycentre, n-body problem with respect to some other body in the solar system, and for quasi-periodic formulation of the restricted n-body problem.

The third chapter will describe the package that was designed as a part of this thesis. It will elaborate the basic structure of the package, installation of the package, and will describe important routines of the package. It will also explain how to implement programs in the package.

The fourth chapter will show some tests and implementation of routines carried out for demonstrating the credibility and the abilities of the package. The routines for coordinate transformation, vector fields, and variational equations have been tested and the results and the steps for these steps have been elaborated in this chapter.

The fifth chapter will elaborate two primary applications of the package. One will be to implement the coefficients of the Lagrangian and Hamiltonian formulation of the model of the solar-system, particularly, the restricted n-body problem. The other application is to perform trajectory refinement using the concept of parallel shooting.

Some of the nomenclature we use in the course of this thesis is as follows:

Barycentre will be used to refer to the centre of the mass of a system. It will have some specific usages. Solar-system barycentre will mean the centre of mass of the whole solar system. On the other hand, a planet barycentre will be used to refer to the centre of mass of the system with that specific planet and all its satellites. For example, Jupiter barycentre would imply that we are talking about the centre of mass of the whole Jovian system i.e. Jupiter and all its satellites. One other usage of the word would explicitly mention two bodies. For example, Saturn-Titan barycentre means the centre of mass of just Saturn and Titan system.

The term RTBP will be occasionally used in this thesis. It stands for Restricted Three Body Problem. RTBP system will refer to the primary and secondary body in such a system. More details are elaborated in Chapter 1.

At the end of this thesis, a robust package has been developed which can be used to implement numerical computations in the solar-system. It has capability to perform orbit refinement using parallel shooting. Also, the variational equations have been implemented and tested and can be used by the user.

Chapter 1

EQUATIONS OF MOTION

1.1 Introduction

The purpose of this chapter is to formulate and introduce the basic equations required to define motion of bodies in a gravitational field. Also, this chapter will focus on definitions of reference frames to be used throughout this thesis and will elaborate how to transform between them. The following subsections will define the equations of n-body problem and will formulate the equations for restricted n-body problem. Before proceeding with the elaboration of equations of motion, we need to define different reference systems and the transformation between them.

1.2 Reference Systems

Reference frames can reduce the complexity in two ways. One, if the reference frame is closer to the spacecraft, for which we are performing the calculations, it can reduce the size of the numerical computations, and, two, if the reference frame is fixed in space and not rotating it can offer advantages in terms of reduction of complexity of a problem by removing dependency on time and making the equations of motion simpler. We will use two primary frames of reference for the present study. The first one is called *Equatorial* coordinate system and the other is called *Adimensional* coordinate system.

1.2.1 Equatorial Reference System

Positions

To define the positions in a reference frame, firstly, the origin of the reference frame has to be defined. The origin of the equatorial reference frame is the centre of the earth. The X-Y plane is defined by the earth's equatorial plane. The positive direction of the Z-axis is defined as the direction to the north pole from the centre of the earth. Positive X direction is defined as the direction of Vernal Equinox. Vernal Equinox is a line drawn from the centre of the earth to the centre of the sun on the first day of the spring. This is denoted by the symbol ♈as this is used to depict the direction of the constellation Aries. A system called the *J2000 system* is what we will be primarily using as our reference system. Vernal Equinox is defined as January 1, 2000 at 12 hrs for J2000 system [2]. The positive direction of the Y-axis is perpendicular to the vernal equinox and is defined by the line joining the centre of the Sun to the centre of the Earth on the first day of winter (December 21 for the *J2000* system). The geocentric equatorial frame is depicted in the Figure 1.1 [1].

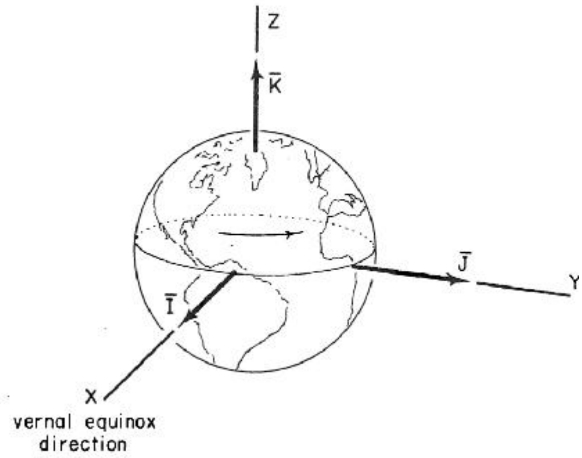


Figure 1.1 Geocentric Equatorial Reference Frame

Time

The time in a reference frame is defined with respect to an initial epoch. The epoch of the *J2000* system is the January 1, 2000 at 00:00 hours (midnight) UTC. This can be represented in terms of Julian Date which by definition is the number of days measured from 4713 B.C but the days start from noon not midnight. J2000 is the Julian Date 2451545 [2], thus, as far as this thesis is concerned, the 'zero' of time in our reference frame is this Julian Date.

1.2.2 Synodical Adimensional Reference System

The other reference frame we will use is called synodical adimensional reference system. This reference system is useful for computation in three body problems and in representing the restricted n-body problem in a form similar to the restricted three body problem. The idea is to remove dimensions of positions and time by expressing them in terms of parameters of the three body system. A restricted n-body system is shown in Figure 1.2. This system has its origin at the centre of mass of the primary and the secondary body. The line joining the primary and the secondary is assumed to be fixed in space. The primary body, with mass m_1, is stationed at coordinates $(\mu, 0, 0)$, where μ is given by Equation (1.1). The secondary body, with mass m_2, is stationed at coordinates $(\mu - 1, 0, 0)$. Then, as can be seen, the distance between the primary and secondary is 'one' in this new system. Thus, the positions are adimensionalized by considering the distance between the two bodies as a unit distance.

$$\mu = \frac{m_2}{m_1 + m_2} \tag{1.1}$$

Time

The time in this reference frame is adimensionalized using the sidereal period of the secondary. Thus, 2π unit of time in adimensional reference frame is the time period of the secondary around the primary This has been elaborated further in the subsequent section. The epoch of the time remains the same as the equatorial reference system.

1.3 Reference Frame Transformations

1.3.1 Equatorial to Adimensional Reference Frame

It might be useful during the study of the equations of motion and during the course of numerical computation of orbits to transform coordinates from the equatorial reference frame to the adimensional reference frame or vice-versa. We will discuss the transformations in terms of positions, velocity, acceleration, and time.

Transformation of Time

The relation between time in the equatorial reference frame , denoted by t^*, to the adimensional frame, denoted by *t*, is given by,

$$t = t^* m \tag{1.2}$$

where *m* is the mean motion of the secondary around the primary and is calculated from the Kepler's third law as,

$$m = \sqrt{\frac{\mu_1 + \mu_2}{a_s^3}} \tag{1.3}$$

$$\mu_1 = G m_1 \tag{1.4}$$

$$\mu_2 = G m_2 \tag{1.5}$$

where a_s is the semi major axis of the orbit of the secondary around the primary and G is the universal gravitational constant. Also, it is important to state the notations being used to define derivatives with respect to time. For any function *f*, the symbol f' will be used in this thesis to represent the differentiation with respect to time in the equatorial reference frame. The symbol $\dot{f}$ will be used to represent the differentiation with respect to adimensional time. Thus,

$$f' = \frac{df}{dt^*} = \frac{d}{dt}\frac{dt}{dt^*} f = \dot{f}\ m \tag{1.6}$$

$$f'' = \frac{d^2}{d(t^*)^2} f = \ddot{f}\ m^2 \tag{1.7}$$

$$f''' = \dddot{f}\ m^3 \tag{1.8}$$

Transformation of Positions, Velocity, and Accelerations

The transformation of positions is described in [3] as,

$$\vec{R} = kC\vec{a} + \vec{b} \tag{1.9}$$

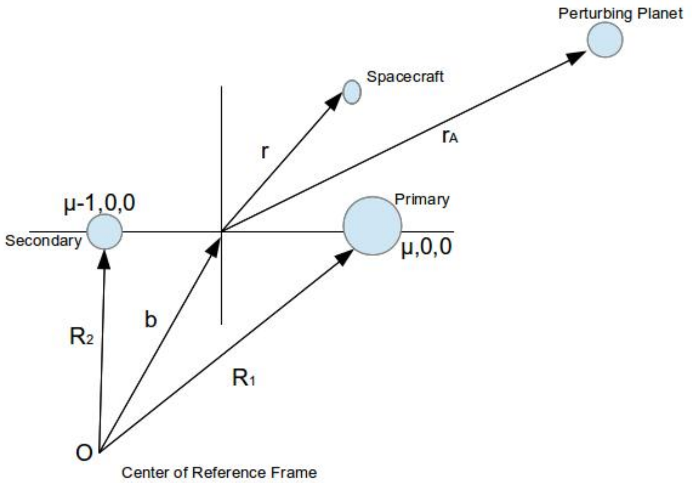


Figure 1.2 Restricted N-Body Problem in Adimensional Coordinates

where k is the change of scale factor, C is an orthogonal matrix, and $\vec{b}$ is the translation vector. The translation vector is nothing but the position of the barycentre of the system from the origin of the equatorial reference frame. The vector $\vec{R}$ is the coordinates of position in the equatorial reference frame and $\vec{a}$ is the coordinates in the adimensional reference frame. C is a 3 by 3 matrix and can be thought to be composed of three column vectors $\vec{C_1}, \vec{C_2}, \vec{C_3}$ which are described as,

$$C = \begin{pmatrix} C_1 & C_2 & C_3 \end{pmatrix} \tag{1.10}$$

where,

$$\vec{C_1} = \frac{\vec{R_{21}}}{||R_{21}||_2} \tag{1.11}$$

$$\vec{C_3} = \frac{\vec{R_{21}} \times \vec{R_{21}}'}{||\vec{R_{21}} \times \vec{R_{21}}'||_2} \tag{1.12}$$

$$\vec{C_2} = \vec{C_3} \times \vec{C_1} \tag{1.13}$$

and,

$$\vec{R_{21}} = \vec{R_1} - \vec{R_2} \tag{1.14}$$

$$\vec{R_{21}}' = \vec{R_1}' - \vec{R_2}' \tag{1.15}$$

In these equations R_1 is the position of the primary and R_2 is the position of the secondary. The scaling factor k can be calculated as,

$$k = ||\vec{R_{21}}||_2 \tag{1.16}$$

and the translation vector $\vec{b}$ is computed as,

$$\vec{b} = \vec{R_1} + \mu \vec{R_{21}} \tag{1.17}$$

where, μ is given by Equation (1.1).

The transformation of the velocities from the equatorial to adimensional coordinates is computed by differentiating Equation (1.9) with respect to time in the equatorial reference frame and additionally using the transformation of time. It i given as,

$$\dot{\vec{a}} = \frac{C^{-1}\vec{e}\,' + C'^{-1}\vec{e} - k'\,\vec{a}}{k\,m} \tag{1.18}$$

where $\vec{e}$ is given according to the Equation (1.19) as,

$$\vec{e} = \vec{R} - \vec{b} \tag{1.19}$$

The transformation from adimensional velocities to the velocities in the equatorial reference frame is given by Equation (1.20) and is obtained by rearranging Equation (1.18).

$$\vec{R}' = \vec{b}' + k' C\vec{a} + k(C'\vec{a} + m\,C\dot{\vec{a}}) \tag{1.20}$$

The matrix C' is 3 by 3 matrix composed of three vectors as columns given by,

$$C' = \begin{pmatrix} C_1' & C_2' & C_3' \end{pmatrix} \tag{1.21}$$

where,

$$C_1' = \frac{k\,\vec{R_{21}}\,' - k'\,\vec{R_{21}}}{||\vec{R_{21}}||_2^2} \tag{1.22}$$

$$C_3' = \frac{\vec{R_{21}} \times \vec{R_{21}}\,''}{||\vec{R_{21}} \times \vec{R_{21}}\,'||_2} - \frac{\vec{C_3}\,.\,(\vec{R_{21}} \times \vec{R_{21}}\,'')}{||\vec{R_{21}} \times \vec{R_{21}}\,'||_2}\vec{R_{21}} \tag{1.23}$$

$$C_2' = C_3' \times \vec{C_1} + \vec{C_3} \times C_1' \tag{1.24}$$

Here, k' can be obtained by differentiating Equation (1.16) with respect to time t as,

$$k' = \frac{\vec{R_{21}}\,.\,\vec{R_{21}}\,'}{k} \tag{1.25}$$

The transformation of acceleration from equatorial to adimensional coordinates is computed by differentiating the Equation (1.18) with respect to equatorial time and then using information on transformation of time given in Section 1.3.1. The transformation is given by Equation (1.26).

$$\ddot{\vec{a}} = \frac{C^{-1}\vec{e}\,'' - k''\,\vec{a} - 2\,k'\,m\vec{a}\,' - C^{-1}(2\,k'\,\vec{e_a} + k\,[\vec{e_b} + 2\vec{e_d}])}{k\,m^2} \tag{1.26}$$

where,

$$\vec{e''} = \vec{R}'' - \vec{b''} \tag{1.27}$$

$$\vec{e_a} = C' \, \vec{a} \tag{1.28}$$

$$\vec{e_b} = C'' \, \vec{a} \tag{1.29}$$

$$\vec{e_d} = m \, C' \, \dot{\vec{a}} \tag{1.30}$$

The transformation of acceleration from adimensional to the equatorial coordinates is given by Equation (1.31) and is obtained by rearranging Equation (1.26).

$$\vec{R}'' = \vec{b''} + k'' \, C \, \vec{a} + k \, C'' \, \vec{a} + m \, C \, \dot{\vec{a}} + 2 \, (k' \, (C' \, \vec{a} + m \, C \, \dot{\vec{a}}) + k \, m \, C' \, \dot{\vec{a}}) \tag{1.31}$$

The matrix C" is 3 by 3 matrix composed of three vectors as columns given by,

$$C'' = \begin{pmatrix} C_1'' & C_2'' & C_3'' \end{pmatrix} \tag{1.32}$$

and,

$$C_1'' = \frac{-\vec{R_{21}}''}{k} + \frac{2 \, k' \, \vec{R_{21}}'}{\|\vec{R_{21}}\|_2^2} + \left(\frac{k'' \, k - 2 \, k'^2}{\|\vec{R_{21}}\|_2^2 \, k} \right) \vec{R_{21}} \tag{1.33}$$

$$C_3'' = \vec{e_f} - \left(C_3' \, . \, C_3' + C_3 \, . \, \vec{e_f} \right) . \, C_3 \tag{1.34}$$

$$C_2'' = C_3'' \times \vec{C_1} + 2 \, (C_3' \times C_1') + \vec{C_3} \times C_1'' \tag{1.35}$$

where,

$$\vec{e_f} = \frac{\vec{R_{21}}' \times \vec{R_{21}}'' + \vec{R_{21}} \times \vec{R_{21}}''' - 2 \, \dfrac{\vec{C_3} \, . \, (\vec{R_{21}} \times \vec{R_{21}}'')}{\|\vec{R_{21}} \times \vec{R_{21}}'\|_2} (\vec{R_{21}} \times \vec{R_{21}}'')}{\|\vec{R_{21}} \times \vec{R_{21}'}\|_2} \tag{1.36}$$

Here, k'' can be obtained by differentiating Equation (1.25) with respect to time as,

$$k'' = \frac{\vec{R_{21}}' \, . \, \vec{R_{21}}' + \vec{R_{21}} \, . \, \vec{R_{21}}'' - k'^2}{k} \tag{1.37}$$

1.3.2 Transformation of Origin of Equatorial Reference Frames

The equatorial reference frame we have defined is geocentric i.e. centred in Earth. Sometimes it is advantageous to perform integration with reference frames centred in some body other than the Earth. For example, while performing integrations in the Saturn-Titan system it might be more advantageous to use a reference frame centred in either Saturn, Titan, or their barycentre. Doing so will reduce the size of the numbers obtained during integrations. Suppose we have to transform coordinates of a body from a reference frame centred in the body A to those centred in body B. Suppose $\vec{r_A}$

denotes the coordinates of the body with respect to the body A and $\vec{r_B}$ denotes the coordinates of the body with respect to the body B. Also, say the vector $\vec{r_{AB}}$ denotes the vector from body A to body B then,

$$\vec{r_B} = \vec{r_A} - \vec{r_{AB}} \tag{1.38}$$

The idea is depicted in Figure 1.3. It is also important to note that during such transformation the directions of all the axis are fixed otherwise rotation of coordinates must be performed. So effectively, only the origin of the reference frame is being transformed.

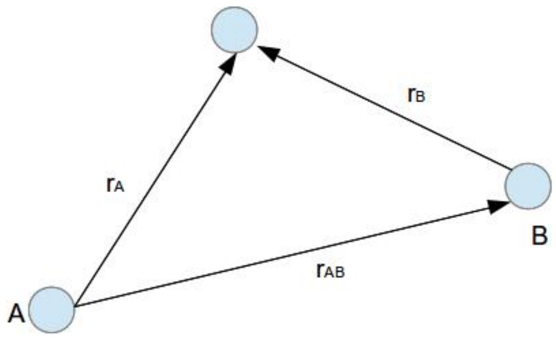


Figure 1.3 Transformation of Origin of Equatorial Reference Frame

1.4 The Restricted N-Body Problem

This section defines the equations to predict the motion of a small body moving in a gravitational field being created by *n-1* number of bodies; a problem referred to as the restricted n-body problem. The restricted n-body problem finds applications in prediction of motion of satellites, spacecrafts, and other small objects in the solar system such as comets and asteroids. Before we generate equations for the restricted n-body problem, first, we need to define the classic n-body problem. The n-Body problem is the prediction of motion of a body in a gravitational vector field produce by *n* bodies. According to the Newton's second law motion, the acceleration of a body *i* moving in such a gravitational field is given by,

$$\ddot{\vec{r_i}} = \sum_{j=1, j\neq i}^{n} \frac{G\, m_j\, \vec{r_{ij}}}{r_{ij}^3} \tag{1.39}$$

where,

$$\vec{r_{ij}} = \vec{r_j} - \vec{r_i} \tag{1.40}$$

$$r_{ij} = [(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2]^{1/2} \tag{1.41}$$

The vectors $\vec{r_i}$ and the vectors $\vec{r_j}$ are referenced with respect to the solar system barycentre at a particular epoch. Thus, the equation (1.39) gives the motion of the body with respect to the solar system barycentre. It can be seen that the Equation (1.39) can be written for a set of *n* different bodies. If we assume that the positions, velocities, and accelerations of all bodies but one are known and that the body for which we do not know these values is very small and cannot influence the motion of the other *n-1* bodies, then, Equation (1.39) will just give one equation, for our target body. This is called as the restricted n-body problem and it finds applications in the design of trajectories of spacecrafts and satellites.

As stated earlier sometimes there might arise a need to see the motion of the body with respect to some other body in the solar system. In this case the Equation (1.39) must be changed. If we want to see the motion of a body *i* with respect to the body *j* we first compute the accelerations of these bodies with respect to solar system barycentre from Equations (1.39) which will be given as,

$$\ddot{\vec{r}_i} = \sum_{k=1,k\neq i}^{n} \frac{G\, m_k\, \vec{r_{ik}}}{r_{ik}^3} = \frac{G\, m_j\, \vec{r_{ij}}}{r_{ij}^3} + \sum_{k=1,k\neq i,j}^{n} \frac{G\, m_k\, \vec{r_{ik}}}{r_{ik}^3} \tag{1.42}$$

and,

$$\ddot{\vec{r}_j} = \sum_{k=1,k\neq j}^{n} \frac{G\, m_k\, \vec{r_{jk}}}{r_{jk}^3} = \frac{G\, m_i\, \vec{r_{ji}}}{r_{ji}^3} + \sum_{k=1,k\neq i,j}^{n} \frac{G\, m_k\, \vec{r_{ik}}}{r_{ik}^3} \tag{1.43}$$

Now, to find the acceleration of body *i* with respect to *j* we subtract Equations (1.42) and (1.43) and we obtain

$$\ddot{\vec{r_{ij}}} = -\frac{G(m_i + m_j)\, \vec{r_{ij}}}{r_{ij}^3} + \sum_{k=1,k\neq i,j}^{n} G\, m_k \left(\frac{\vec{r_{ik}}}{r_{ik}^3} - \frac{\vec{r_{jk}}}{r_{jk}^3}\right) \tag{1.44}$$

In this equation we have also used the fact that $\vec{r_{ij}} = -\vec{r_{ji}}$. Since we assume that the mass of the body *i* is very small as compared to other bodies in the system, $m_i + m_j$ can be approximated as m_j. Also, since we are trying to write equations with respect to the body *j* it is better to obtain positions of every other body with respect to *j* rather than the solar system barycentre. Thus, if every vector is expressed with respect to the body *j* we can re-write the Equation (1.44) as,

$$\ddot{\vec{r}_i} = -\frac{G\, m_j\, \vec{r_i}}{r_i^3} + \sum_{k=1,k\neq i,j}^{n} G\, m_k \left(\frac{\vec{r_{ik}}}{r_{ik}^3} - \frac{\vec{r_k}}{r_k^3}\right) \tag{1.45}$$

It is very important to note that all the vectors in Equation (1.45) are with respect to the body *j*. For all practical purposes this thesis will use the Equation (1.45) to predict the motion of a small body such as a satellite in the restricted n-body problem.

1.5 The Restricted Three Body Problem

The aim of this section is to introduce the restricted three body problem and to introduce tools that will be required to numerically analyse the said problem. Consider two bodies of masses m_1 and m_2 that are moving only under the influence of their mutual gravitational attraction in a circular orbit. We call the heavier of these two bodies as primary and the other as secondary. Suppose we have a third body with a mass m_3, which is small enough so that it does not affect the motion of the primary and secondary bodies, moving under the influence of the gravity of the primary and the secondary.

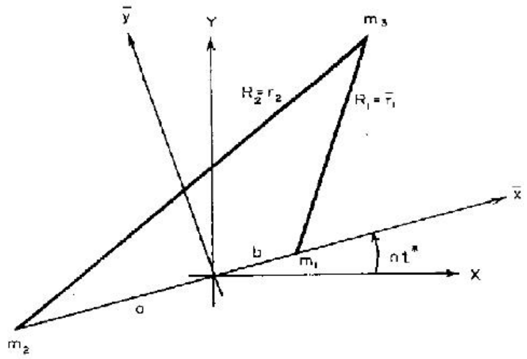


Figure 1.4 Depiction of Restricted Three Body System

Figure 1.4 [5] shows the projection of a three body system on XY plane. We consider two separate reference frames. One, the *XYZ* inertial frame which is fixed in space (also referred to as the sidereal frame), and two, the $\bar{X}\bar{Y}\bar{Z}$ reference frame which is rotating with the system (also referred to as the synodic frame). The Z axis of both the frames can be assumed to be projecting out of the plane and will be the same as the frame is rotating about the Z-axis. Assume that the mass m_1 is located at the coordinates (in the inertial reference frame) (X_1, Y_1, Z_1) and the mass $m_2 (m_1 > m_2)$ is located at (X_2, Y_2, Z_2). Masses m_1 and m_2 are separated by distance *l*. The distance of mass m_2 from the origin of the coordinate system is *a* and the distance of m_1 is *b*. Assume that the angular velocity of the masses around the centre of the coordinate axis is $\omega = n$. The gravitational attraction between the bodies must be balanced by the centrifugal force. Therefore,

$$\frac{G\, m_1\, m_2}{l^2} = m_1 \omega^2 b \tag{1.46}$$

where G is the universal gravitational constant. The equation of motion of the third body i.e. m_3 can be derived using the newton's second law of motion.

$$\frac{d^2 X}{dt^{*2}} = \frac{\partial F}{\partial X}\ ,\ \frac{d^2 Y}{dt^{*2}} = \frac{\partial F}{\partial Y} \text{ and } \frac{d^2 Z}{dt^{*2}} = \frac{\partial F}{\partial Z} \tag{1.47}$$

It is important to point out that t^* represents the time in the inertial frame. Here, F is the force function which is given as [5],

$$F = G\left(\frac{m_1}{R_1} + \frac{m_2}{R_2}\right) \tag{1.48}$$

where, R_1 and R_2 are given by,

$$R_1 = \sqrt{(X - X_1)^2 + (Y - Y_1)^2 + (Z - Z_1)^2} \tag{1.49}$$

$$R_2 = \sqrt{(X - X_2)^2 + (Y - Y_2)^2 + (Z - Z_2)^2} \tag{1.50}$$

Also, X_1,Y_1 and X_2, Y_2 can be expressed in terms of the angular velocity as,

$$X_1 = b\,cos\,\omega t^*, Y_1 = b\,sin\,\omega t^* \tag{1.51}$$

$$X_2 = -a\,cos\,\omega t^*, Y_2 = -a\,sin\,\omega t^* \tag{1.52}$$

Putting the values from Equations (1.48) to (1.52) in Equation (1.47) we obtain the acceleration of the body m_3 in the inertial frame *XY* as [5],

$$\frac{d^2X}{dt^{*2}} = -G\left(\frac{m_1\,(X - b\,cos\,\omega t^*)}{R_1^3} + \frac{m_2\,(X + a\,cos\,\omega t^*)}{R_2^3}\right) \tag{1.53}$$

$$\frac{d^2Y}{dt^{*2}} = -G\left(\frac{m_1\,(Y - b\,sin\,\omega t^*)}{R_1^3} + \frac{m_2\,(Y + b\,sin\,\omega t^*)}{R_2^3}\right) \tag{1.54}$$

$$\frac{d^2Z}{dt^{*2}} = -G\left(\frac{m_1\,Z_1}{R_1^3} + \frac{m_2\,Z_2}{R_2^3}\right) \tag{1.55}$$

It is easy to understand that since the line connecting the masses m_1 and m_2 is rotating in space, the coordinates of the mass m_3, when expressed in the inertial reference frame XYZ, will be dependent on time because of the angular velocity of the system. This can be confirmed from the Equations (1.53) and (1.54). It makes more sense to remove this time dependency by assuming that the line connecting the masses m_1 and m_2 is fixed and to consider the reference frame $\bar{X}\bar{Y}\bar{Z}$.

1.5.1 Equations in Synodic Coordinates

This section will formulate the equations of motion of the restricted three body problem in synodic coordinate system. Following the approach suggested in [5] we use complex variables to transform coordinates from sidereal to synodic reference frame.

$$P = p\,e^{i\omega t^*} \tag{1.56}$$

where,

$$p = \bar{X} + i\bar{Y} \tag{1.57}$$

$$P = X + iY \tag{1.58}$$

Also R_1 and R_2, the distances of the primary and secondary respectively, can be expressed as,

$$R_1 = ||\,P - P_1|| \tag{1.59}$$

$$R_2 = ||\,P - P_2|| \tag{1.60}$$

The Equation (1.51) and (1.52) can be re-written in terms of these new complex coordinate system as,

$$P_1 = b\,c^{i\omega t^*} \tag{1.61}$$

$$P_2 = -a\, c^{i\omega t^*} \quad \textbf{(1.62)}$$

Now, R_1 and R_2 can be expressed as,

$$R_1 = \|p - b\| = \sqrt{(\bar{X} - b)^2 + \bar{y}^2} \quad \textbf{(1.63)}$$

$$R_2 = \|p + a\| = \sqrt{(\bar{X} + a)^2 + \bar{y}^2} \quad \textbf{(1.64)}$$

Then, the equations of motion in complex system becomes,

$$\frac{d^2P}{dt^{*2}} = \left(\frac{d^2p}{dt^{*2}} + 2i\,\omega\frac{dz}{dt^*} - \omega^2 p\right)e^{iwt^*} \quad \textbf{(1.65)}$$

Using this equation with equations from (1.56) to (1.64) and rearranging we can get the complex form of the equations of motion in a circular rotating frame [5],

$$\frac{d^2p}{dt^{*2}} + 2i\,\omega\frac{dz}{dt^*} - \omega^2 p = -G\left(m_1\,\frac{(p-b)}{\|p-b\|^3} + m_2\,\frac{(p+a)}{\|p+a\|^3}\right) \quad \textbf{(1.66)}$$

Now equating the real and complex parts we can express this equation as,

$$\frac{d^2\bar{x}}{dt^*} - 2\,\omega\frac{d\bar{y}}{dt^*} - \omega^2\bar{x} = -G\left(m_1\,\frac{\bar{x}-b}{\bar{r}_1^3} + m_2\,\frac{\bar{x}+a}{\bar{r}_2^3}\right) \quad \textbf{(1.67)}$$

$$\frac{d^2\bar{y}}{dt^*} + 2\,\omega\frac{d\bar{x}}{dt^*} - \omega^2\bar{y} = -G\left(m_1\,\frac{\bar{y}}{\bar{r}_1^3} + m_2\,\frac{\bar{y}}{\bar{r}_2^3}\right) \quad \textbf{(1.68)}$$

This can be expressed in terms of the Force function F^* in synodical coordinates as,

$$\frac{d^2\bar{x}}{dt^*} - 2\,\omega\frac{d\bar{y}}{dt^*} = \frac{\partial F^*}{\partial\bar{x}} \quad \textbf{(1.69)}$$

$$\frac{d^2\bar{y}}{dt^*} + 2\,\omega\frac{d\bar{x}}{dt^*} = \frac{\partial F^*}{\partial\bar{y}} \quad \textbf{(1.70)}$$

where,

$$F^* = \frac{\omega^2}{2}(\bar{x}^2 + \bar{y}^2) + G\left(\frac{m_1}{\bar{r}_1} + \frac{m_2}{\bar{r}_2}\right) \quad \textbf{(1.71)}$$

By now, we have only expressed two components of the equations of motion. The z component is easy to transform since $Z = \bar{Z}$. Thus,

$$\frac{d^2\bar{z}}{dt^*} = \frac{\partial F^*}{\partial\bar{z}} \quad \textbf{(1.72)}$$

Equations (1.69), (1.70), and (1.72) are the final equations of motion in the synodical reference frame.

1.5.2 Restricted Three Body Problem in Adimensional Coordinates

It is interesting to write the equations of the restricted three body problem in adimensional synodic reference frame and to see how much simplicity such a process can offer. Firstly, we will define some parameters to remove the dimensions of the problem:

$$\begin{gathered} x = \frac{\bar{x}}{l}\ , y = \frac{\bar{y}}{l},\ \ t = \omega t^*,\ \ r_1 = \frac{\bar{r}_1}{l} \\ r_2 = \frac{\bar{r}_2}{l},\ \ \mu = \frac{m_2}{m_1 + m_2} \end{gathered} \tag{1.73}$$

Using these parameters Equations (1.69), (1.70), and (1.72) can be re-written in adimensional form as [4],

$$\ddot{x} - 2\dot{y} = \bar{\Omega}_x \tag{1.74}$$

$$\ddot{y} + 2\dot{x} = \bar{\Omega}_y \tag{1.75}$$

$$\ddot{z} = \bar{\Omega}_z \tag{1.76}$$

where,

$$\Omega = \frac{1}{2}(x^2 + y^2) + \frac{1-\mu}{r_1} + \frac{\mu}{r_2} + \frac{\mu(1-\mu)}{2} \tag{1.77}$$

and,

$$r_1^2 = (x - 1 + \mu)^2 + y^2 + z^2 \tag{1.78}$$

$$r_2^2 = (x + \mu)^2 + y^2 + z^2 \tag{1.79}$$

These equations are the well known form of the restricted three body problem. Although we will not be studying this problem explicitly, the derivation of these equations have been presented because we will try to express the restricted n-body problem in a form similar to these equations.

1.6 Lagrangian Formulation of the Restricted N-Body Problem

A general discussion of Lagrangian Mechanics is elaborated in Annexure A. The Lagrangian formulation of Equation (1.39) can be expressed as,

$$L = \frac{1}{2}\dot{\vec{R}}\dot{\vec{R}} + \sum_{A\epsilon(S,E,M,P...,P_k),A\neq m_1,m_2} \frac{G\,m_A}{||\vec{R} - \vec{R_A}||_2} \tag{1.80}$$

This equation contains two terms. The first term contains $\dot{\vec{R}}$, which is the velocity of the body (eg. a spacecraft) for which we are writing this equation. The second term is the combined perturbation of all the other bodies in the solar system except the primary and the secondary body. Also, G is the gravitational constant, m_A is the mass of the perturbing body, $\vec{R}$ is the position of the spacecraft, and $\vec{R_A}$ is the position of the perturbing body.

It is important to note that this equation is in equatorial reference frame. If we want to obtain this expression in adimensional coordinates RTBP like system, i.e., selecting a primary and secondary) then we have to use coordinate transformation. Suppose that the adimensional coordinates are represented by the vector $\vec{r}$, then using Equation (1.9) we obtain,

$$\vec{R} = \vec{B} + k\,C\,\vec{r} \tag{1.81}$$

where,

$$\vec{r} = X\,\hat{\imath} + Y\,\hat{\jmath} + Z\,\hat{k} \tag{1.82}$$

The vector $\vec{B}$ is given by Equation (1.17).

Using the equations above we obtain,

$$\vec{R}'.\vec{R}' = \vec{B}'.\vec{B}' + 2k'\vec{B}'.\vec{S} + 2\,k\vec{B}'.\vec{S}' + (k')^2\vec{r}.\vec{r} + 2\,k\,k'\vec{S}.\vec{S}' + 2\,k^2\vec{S}'.\vec{S}' \tag{1.83}$$

where,

$$\vec{S} = C\,\vec{r} \tag{1.84}$$

and the matrix C is given by Equation (1.10). Also,

$$\|R - R_A\|_2 = k\,\|r - r_A\|_2 \tag{1.85}$$

Here, k is given from Equation (1.16). Using the equations above we can express the Lagrangian formulation of Equation (1.39) in adimensional coordinates as,

$$\begin{aligned} L(\vec{r},\vec{r}',t^*) = {} & \frac{1}{2}\vec{B}'.\vec{B}' + k'\vec{B}'.\vec{S} + k\,\vec{B}'.\vec{S}' + \frac{(k')^2}{2}\vec{r}.\vec{r} + k\,k'\vec{S}.\vec{S}' + \frac{k^2}{2}\vec{S}'.\vec{S}' \\ & + \frac{G\,m_1}{k\sqrt{(X-\mu)^2 + Y^2 + Z^2}} + \frac{G\,m_2}{k\sqrt{(X-\mu+1)^2 + Y^2 + Z^2}} \\ & + \sum_{A\epsilon(S,E,M,P\ldots,P_k),A\neq m_1,m_2} \frac{G\,m_A}{k\,\|r - r_A\|_2} \end{aligned} \tag{1.86}$$

It is important to note that Equation (1.86) is in terms of time in equatorial reference frame. We need to transform the time into the adimensional time as well. Using information in Section 1.3.1 we obtain the Lagrangian formulation of the restricted n-body problem in adimensional time t as,

$$\begin{aligned} L(\vec{r},\vec{r}',t) = {} & \frac{1}{a^2}\left(\frac{1}{2}\dot{\vec{B}}.\dot{\vec{B}} + \dot{k}\,\dot{\vec{B}}.\vec{S} + k\,\dot{\vec{B}}.\dot{\vec{S}} + \frac{\dot{k}^2}{2}\vec{r}.\vec{r} + k\,\dot{k}\,\vec{S}.\dot{\vec{S}} + \frac{k^2}{2}\dot{\vec{S}}.\dot{\vec{S}}\right) \\ & + \frac{a}{k}\left(\frac{G\,m_1}{k\sqrt{(X-\mu)^2 + Y^2 + Z^2}} + \frac{G\,m_2}{k\sqrt{(X-\mu+1)^2 + Y^2 + Z^2}}\right. \\ & \left. + \sum_{A\epsilon(S,E,M,P\ldots,P_k),A\neq m_1,m_2} \frac{G\,m_A}{k\,\|r - r_A\|_2}\right) \end{aligned} \tag{1.87}$$

In this equation *a* is the semi-major axis of the secondary body around the primary. Using Equations (1.81) and (1.84) we obtain,

$$\vec{S} = C_1 X + C_2 Y + C_3 Z \tag{1.88}$$

where, C_1, C_2, and C_3 are column vectors given by Equation (1.11), (1.12), (1.13) respectively. Also, differentiating Equation (1.88) with respect to adimensional time we obtain,

$$\dot{\vec{S}} = \dot{C}_1 X + \dot{C}_2 Y + \dot{C}_3 Z + C_1 \dot{X} + C_2 \dot{Y} + C_3 \dot{Z} \tag{1.89}$$

where, $\dot{C}_1$, $\dot{C}_2$, and $\dot{C}_3$ using Equations (1.22) through (1.24). Finally, using the equations above we can calculate,

$$\dot{k}\,\dot{\vec{B}}.\vec{S} = \dot{k}\,(\dot{B}.C_1)\,X + \dot{k}\,(\dot{B}.\ C_2)Y + \dot{k}\,(\dot{B}.C_3)\,Z \tag{1.90}$$

$$\begin{aligned} k\,\dot{\vec{B}}.\dot{\vec{S}} = (\dot{B}\,.\,\dot{C}_1)\,X + k(\dot{B}\,.\,\dot{C}_2)\,Y + k\,(\dot{B}\,.\,\dot{C}_3)\,Z + k\,(\dot{B}\,.\,C_1)\,\dot{X} \\ +k\,(\dot{B}\,.\,C_2)\,\dot{Y} + k\,(\dot{B}\,.\,C_3)\,\dot{Z} \end{aligned} \tag{1.91}$$

$$\frac{\dot{k}^2}{2}\vec{r}\,.\,\vec{r} = \frac{1}{2}\dot{k}^2X^2 + \frac{1}{2}\dot{k}^2Y^2 + \frac{1}{2}\dot{k}^2Z^2 \tag{1.92}$$

$$k\,\dot{k}\,\vec{S}\,.\,\dot{\vec{S}} = k\,\dot{k}\,X\dot{X} + k\,\dot{k}\,Y\,\dot{Y} + k\,\dot{k}\,Z\,\dot{Z} \tag{1.93}$$

$$\begin{aligned} \frac{k^2}{2}\dot{\vec{S}}.\dot{\vec{S}} = \frac{k^2}{2}\dot{X}^2 + \frac{k^2}{2}\dot{Y}^2 + \frac{k^2}{2}\dot{Z}^2 + \frac{k^2}{2}(\dot{C}_1\,.\,\dot{C}_1)\,X^2 + \frac{k^2}{2}(\dot{C}_2\,.\,\dot{C}_2)\,Y^2 \\ +\frac{k^2}{2}(\dot{C}_3\,.\,\dot{C}_3)\,Z^2 + k^2\,(\dot{C}_1\,.\,\dot{C}_3)\,X\,Z + k^2\,(\dot{C}_1\,.\,C_2)\,(X\,\dot{Y} - \dot{X}\,Y) \\ +k^2\,(\dot{C}_1\,.\,\dot{C}_3)\,X\,Z + k^2\,(\dot{C}_2\,.\,C_3)\,(Y\,\dot{Z} - \dot{Y}\,Z) \end{aligned} \tag{1.94}$$

Substituting values of Equations (1.90) to (1.94) in Equation (1.87) and rearranging the terms we obtain a more organised form of the Lagrangian formulation as,

$$\begin{aligned} L(\vec{r},\dot{\vec{r}},t) = a_1\,(\dot{X}^2 + \dot{Y}^2 + \dot{Z}^2) + a_2\,(X\,\dot{X} + Y\,\dot{Y} + Z\,\dot{Z}) + a_3\,(X\,\dot{Y} + \dot{X}\,Y) + a_4\,(Y\,\dot{Z} - \dot{Y}\,Z) \\ +a_5\,X^2 + a_6\,Y^2 + a_7\,Z^2 + a_8\,X\,Z + a_9\,\dot{X} + a_{10}\,\dot{Y} + a_{11}\,\dot{Z} + a_{12}\,X + a_{13}\,Y + a_{14}\,Z \\ +a_{15}\left(\frac{Gm_1}{k\,\sqrt{(X-\mu)^2 + Y^2 + Z^2}} + \frac{Gm_2}{k\,\sqrt{(X-\mu+1)^2 + Y^2 + Z^2}}\right. \\ \left. + \sum_{A\epsilon(S,E,M,P...,P_k),A\neq m_1,m_2} \frac{Gm_A}{k\|r - r_A\|_2}\right) \end{aligned} \tag{1.95}$$

The coefficients for the equations are given from,

$$a_1 = \frac{1}{2}\left(\frac{k}{a}\right)^2 \tag{1.96}$$

$$a_2 = \frac{k\dot{k}}{a^2} \tag{1.97}$$

$$a_3 = \left(\frac{k}{a}\right)^2 (\dot{C}_1 \, . \, C_2) \tag{1.98}$$

$$a_4 = \left(\frac{k}{a}\right)^2 (\dot{C}_2 \, . \, C_3) \tag{1.99}$$

(1.100)

$$a_5 = \frac{1}{2}\left[\left(\frac{\dot{k}}{a}\right)^2 + \left(\frac{k}{a}\right)^2 (\dot{C}_1 \, . \, \dot{C}_1)\right] \tag{1.101}$$

$$a_6 = \frac{1}{2}\left[\left(\frac{\dot{k}}{a}\right)^2 + \left(\frac{k}{a}\right)^2 (\dot{C}_2 \, . \, \dot{C}_2)\right] \tag{1.102}$$

$$a_7 = \frac{1}{2}\left[\left(\frac{\dot{k}}{a}\right)^2 + \left(\frac{k}{a}\right)^2 (\dot{C}_3 \, . \, \dot{C}_3)\right] \tag{1.103}$$

(1.104)

$$a_8 = \left(\frac{k}{a}\right)^2 (\dot{C}_3 \, . \, \dot{C}_3) \tag{1.105}$$

$$a_9 = \left(\frac{k}{a^2}\right)(\dot{B} \, . \, C_1) \tag{1.106}$$

$$a_{10} = \left(\frac{k}{a^2}\right)(\dot{B} \, . \, C_2) \tag{1.107}$$

$$a_{11} = \left(\frac{k}{a^2}\right)(\dot{B} \, . \, C_3) \tag{1.108}$$

(1.109)

$$a_{12} = \left(\frac{\dot{k}}{a^2}\right)(\dot{B} \, . \, C_1) + \left(\frac{k}{a^2}\right)(\dot{B} \, . \, \dot{C}_1) \tag{1.110}$$

$$a_{13} = \left(\frac{\dot{k}}{a^2}\right)(\dot{B} \, . \, C_2) + \left(\frac{k}{a^2}\right)(\dot{B} \, . \, \dot{C}_2) \tag{1.111}$$

$$a_{14} = \left(\frac{\dot{k}}{a^2}\right)(\dot{B} \, . \, C_3) + \left(\frac{k}{a^2}\right)(\dot{B} \, . \, \dot{C}_3) \tag{1.112}$$

$$a_{15} = \frac{a}{k} \tag{1.113}$$

These coefficients are quasi-periodic functions of time. This quasi-periodicity is for hundreds of thousands of years, which is ”short” a period of time considering the age of the universe.

1.7 Hamiltonian Formulation of the Restricted N-Body Problem

This section elaborates the Hamiltonian formulation for restricted n-body problem. According to information in Section A.4 we have,

$$H = \dot{X}P_X + \dot{Y}P_Y + \dot{Z}P_Z - L \tag{1.114}$$

For restricted n-body problem (in adimensional coordinates and time) we have the momentum given by Equation (A.20) as,

$$P_X = 2\,a_1\,\dot{X} + a_2\,X - a_3\,Y + a_9 \tag{1.115}$$
$$P_Y = 2\,a_1\,\dot{Y} + a_2\,Y + a_3\,X - a_4\,Z + a_{10} \tag{1.116}$$
$$P_Z = 2\,a_1\,\dot{Z} + a_2\,Z + a_4\,Y + a_{11} \tag{1.117}$$

Re-arranging these equations we obtain,

$$\dot{X} = \frac{P_X}{2\,a_1} - \frac{a_2\,X}{2\,a_1} + \frac{a_3\,Y}{2\,a_1} - \frac{a_9}{2\,a_1} \tag{1.118}$$
$$\dot{Y} = \frac{P_Y}{2\,a_1} - \frac{a_3\,X}{2\,a_1} - \frac{a_2\,Y}{2\,a_1} + \frac{a_4\,Z}{2\,a_1} - \frac{a_{10}}{2\,a_1} \tag{1.119}$$
$$\dot{Z} = \frac{P_Z}{2\,a_1} - \frac{a_4\,Y}{2\,a_1} - \frac{a_2\,Z}{2\,a_1} - \frac{a_{11}}{2\,a_1} \tag{1.120}$$

The Hamiltonian can be obtained substituting values from these equations into Equation (1.114). By doing so we obtain the Hamiltonian as the Equation (1.121). The coefficients required for this equation are dependent on the coefficients of Lagrangian and are given by Equations (1.122) to (1.136)

$$\begin{aligned} H(\vec{r}, P, t) = {} & b_1\,(P_X^2 + P_Y^2 + P_Z^2) + b_2\,(X\,P_X + Y\,P_Y + Z\,P_Z) + b_3\,(YP_X - XP_Y) \\ & + b_4\,(Z\,P_Y - Y\,P_Z) + b_5\,X^2 + b_6\,Y^2 + b_7\,Z^2 + b_8\,X\,Z + b_9\,P_X + b_{10}\,P_Y \\ & + b_{11}\,P_Z + b_{12}\,X + b_{13}\,Y + b_{14}\,Z \\ & + b_{15}\left(\frac{1-\mu}{\sqrt{(X-\mu)^2 + Y^2 + Z^2}} + \frac{\mu}{\sqrt{(X-\mu+1)^2 + Y^2 + Z^2}} \right. \\ & \left. + \sum_{A\epsilon(S,E,M,P\ldots,P_k),A\neq m_1,m_2} \frac{G\,m_A}{k\,\|r - r_A\|_2}\right) \end{aligned} \tag{1.121}$$

$$b_1 = \frac{1}{4\,a_1} \tag{1.122}$$
$$b_2 = \frac{-\,a_2}{2\,a_1} \tag{1.123}$$
$$b_3 = \frac{a_3}{2\,a_1} \tag{1.124}$$
$$b_4 = \frac{a_4}{2\,a_1} \tag{1.125}$$
$$b_5 = \frac{a_2^2 + a_3^2}{4a_1} - a_5 \tag{1.126}$$
$$b_6 = \frac{a_2^2 + a_3^2 + a_4^2}{4\,a_1} - a_6 \tag{1.127}$$

$$b_7 = \frac{a_2^2 + a_4^2}{4\,a_1} - a_7 \tag{1.128}$$

$$b_8 = \frac{a_3\,a_4}{2\,a_1} - a_8 \tag{1.129}$$

$$b_9 = \frac{-\,a_9}{2\,a_1} \tag{1.130}$$

$$b_{10} = \frac{-\,a_{10}}{2\,a_1} \tag{1.131}$$

$$b_{11} = \frac{-\,a_{11}}{2\,a_1} \tag{1.132}$$

$$b_{12} = \frac{a_2 a_9}{2\,a_1} + \frac{a_3\,a_{10}}{2\,a_1} - a_{12} \tag{1.133}$$

$$b_{13} = \frac{-\,a_3\,a_9}{2\,a_1} + \frac{a_2\,a_{10}}{2\,a_1} + \frac{a_4\,a_{11}}{2\,a_1} - a_{13} \tag{1.134}$$

$$b_{14} = \frac{-\,a_4\,a_{10}}{2\,a_1} + \frac{a_2\,a_{11}}{2\,a_1} - a_{14} \tag{1.135}$$

$$b_{15} = -a_{15} \tag{1.136}$$

It can be seen with simple calculations that,

$$b_5 = b_6 = b_7 = b_8 = b_{12} = b_{13} = b_{14} = 0 \tag{1.137}$$

Again, these coefficients are quasi-periodic functions of time like the coefficients of the Lagrangian.

1.8 Quasi-Periodic Formulation of the Restricted N-Body Problem

The equations of motion for restricted n-body problem in synodical adimensional coordinates are obtained using the Equation (A.18) for the three components i.e. x, y, and z. Thus, we formulate the equations of motion as,

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{x}} - \frac{\partial L}{\partial x} = 0 \tag{1.138}$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{y}} - \frac{\partial L}{\partial y} = 0 \tag{1.139}$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{z}} - \frac{\partial L}{\partial z} = 0 \tag{1.140}$$

Performing these computations and rearranging we get the quasi-periodic semi-analytical RTBP like formulation for restricted n-body problem in the synodical adimensional reference frame as,

$$\ddot{x} = c_1 + c_4\,\dot{x} + c_5\,\dot{y} + c_7\,x + c_8\,y + c_9\,z + c_{13}\,\frac{\partial \Omega}{\partial x} \tag{1.141}$$

$$\ddot{y} = c_2 - c_5\,\dot{x} + c_4\,\dot{y} + c_8\,x + c_{10}\,y + c_{11}\,z + c_{12}\,\frac{\partial\Omega}{\partial y} \tag{1.142}$$

$$\ddot{z} = c_3 - c_6\,\dot{y} + c_4\,\dot{z} + c_9\,x - c_{11}\,y + c_{12}\,z + c_{13}\,\frac{\partial\Omega}{\partial z} \tag{1.143}$$

where,

$$\begin{aligned}\Omega = &\frac{1-\mu}{\sqrt{(x-\mu)^2+y^2+z^2}} + \frac{\mu}{\sqrt{(x-\mu+1)^2+y^2+z^2}} \\ &+ \sum_{A\epsilon(S,E,M,P\ldots,P_k),A\neq m_1,m_2} \frac{\mu_A}{\|\vec{r}-\vec{r_A}\|_2}\end{aligned} \tag{1.144}$$

In equation (1.144) the vectors $\vec{r}$ and $\vec{r_A}$ are given by,

$$\vec{r} = x\,\hat{i} + y\,\hat{j} + z\,\hat{k} \tag{1.145}$$

$$\vec{r_A} = x_A\,\hat{i} + y_A\,\hat{j} + z_A\,\hat{k} \tag{1.146}$$

It can be seen that these equations are similar in form to the equations of restricted three body motion in the synodic reference frame. The coefficients are given by,

$$c_1 = \frac{-(\ddot{\vec{B}}\,.\,\vec{C_1})}{k} \tag{1.147}$$

$$c_2 = \frac{-(\ddot{\vec{B}}\,.\,\vec{C_2})}{k} \tag{1.148}$$

$$c_3 = \frac{-(\ddot{\vec{B}}\,.\,\vec{C_3})}{k} \tag{1.149}$$

$$c_4 = \frac{-2\,\dot{k}}{k} \tag{1.150}$$

$$c_5 = 2\,(\dot{\vec{C_1}}\,.\,\vec{C_2}) \tag{1.151}$$

$$c_6 = 2\,(\dot{\vec{C_2}}\,.\,\vec{C_3}) \tag{1.152}$$

$$c_7 = (\dot{\vec{C_1}}\,.\,\dot{\vec{C_1}}) - \frac{\ddot{k}}{k} \tag{1.153}$$

$$c_8 = \frac{2\,\dot{k}\,(\dot{\vec{C_1}}\,.\,\vec{C_2})}{k} + (\ddot{\vec{C_1}}\,.\,\vec{C_2}) \tag{1.154}$$

$$c_9 = (\dot{\vec{C_1}}\,.\,\dot{\vec{C_3}}) \tag{1.155}$$

$$c_{10} = (\dot{\vec{C_2}}\,.\,\dot{\vec{C_2}}) - \frac{\ddot{k}}{k} \tag{1.156}$$

$$c_{11} = \frac{2\,\dot{k}\,(\dot{\vec{C_2}}\,.\,\vec{C_3})}{k} + (\ddot{\vec{C_2}}\,.\,\vec{C_3}) \tag{1.157}$$

$$c_{12} = (\dot{\vec{C_3}}\,.\,\dot{\vec{C_3}}) - \frac{\ddot{k}}{k} \tag{1.158}$$

$$c_{13} = \left(\frac{a}{k}\right)^3 \tag{1.159}$$

It is important to point out that these coefficients are dependent on time and are quasi periodic and thus Fourier analysis can be performed. After the Fourier Analysis the expansions can be cut and only the most relevant terms can be kept and in this way we can obtain suitable semi-analytical models that approximate up to a desired precision the complete model of the system.

Chapter 2

VARIATIONAL EQUATIONS

2.1 Variational Equations

Variational equations are important as they express the change in the final state of a system when a change is applied to initial state of the system. The study of these equations is very important as they can be used to analyse the change in orbit of a body because of a small perturbation in the initial position and velocities. The purpose of this chapter is to introduce the mathematical background of variational equations and to formulate the variational equations for three primary vector fields:

1. Restricted n-body problem described with respect to the solar system barycentre
2. Restricted n-body problem described with respect to some particular body in the solar system
3. Quasi-periodic formulation of restricted n-body problem

2.2 State Transition Matrix

The first step before considering the variational equations is to understand the concept of State Transition Matrix. Let $\phi_t(x_0)$ represent the state of a dynamical system at a time *t* such that the state of the system at *t=0* was x_0. It is important to point out that the state of the dynamical system at *t=0* will be the initial point x_0. This can be stated as,

$$\phi_0(x_0) = x_0 \tag{2.1}$$

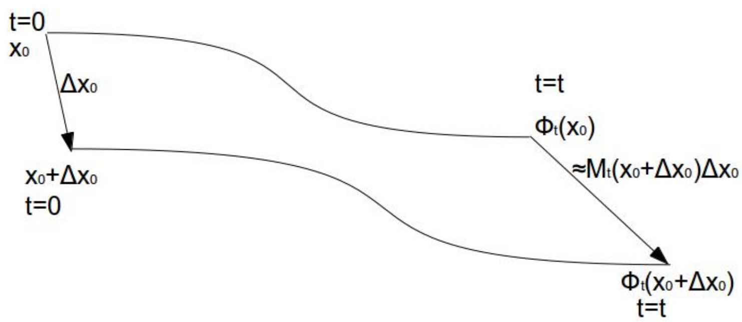


Figure 2.1 State Transition Matrix

Now if the initial state was changed a little by say Δx_0 then the state at a time *t* will be $\phi_t(x_0 + \Delta x_0)$. This can be expanded binomially as,

$$\phi_t(x_0 + \Delta x_0) = \phi_t(x_0) + D_x\phi_t(x_0)\Delta x_0 + \cdots \tag{2.2}$$

By dropping the higher order terms this can be approximated as,

$$\phi_t(x_0 + \Delta x_0) - \phi_t(x_0) \approx D_x\phi_t(x_0)\Delta x_0 \tag{2.3}$$

The matrix *M* is called state transition matrix and is given by,

$$M_t(x_0) = D_x\phi_t(x_0) \tag{2.4}$$

The concept of Equation (2.3) is represented in Figure 2.1. It is easy to see that,

$$M_0(\phi_t(x_0)) = I \tag{2.5}$$

where, *I* represents identity matrix.

2.3 Differential of State Transition Matrix

Consider a dynamical system represented by,

$$\dot{X} = f(X) \tag{2.6}$$

Then it is clear that the state of the dynamical system $\phi_t(x)$ will be a solution to Equation (2.6). Thus,

$$\frac{d(\phi_t(x))}{dt} = f(\phi_t(x)) \tag{2.7}$$

Differentiating (2.6) with respect to x we get,

$$D_x\Big(\frac{d(\phi_t(x))}{dt}\Big) = D_x\Big(f(\phi_t(x))\Big)\,D_x(\phi_t(x)) \tag{2.8}$$

This equation can be re-written as,

$$\frac{d}{dt}(D_x\phi_t(x)) = D_x\Big(f(\phi_t(x))\Big)\,D_x(\phi_t(x)) \tag{2.9}$$

$D_x(\phi_t(x))$ is the state transition matrix as per Equation (2.4). Thus equation (2.9) can be rewritten as,

$$\dot{M} = D_x\Big(f(\phi_t(x))\Big)M \tag{2.10}$$

We call the matrix *D* as *Variational matrix* and it is given as,

$$D = D_x\Big(f(\phi_t(x))\Big) \tag{2.11}$$

2.4 Computation of Variational Matrix

2.4.1 Autonomous system

Consider the state of an autonomous system represented by the vector X such that,

$$X = (x, y, z, \dot{x}, \dot{y}, \dot{z})^T \tag{2.12}$$

then the differential of the X with respect of time is given by,

$$\dot{X} = (\dot{x}, \dot{y}, \dot{z}, \ddot{x}, \ddot{y}, \ddot{z})^T \tag{2.13}$$

For this case, the variational matrix D is given by Equation (2.14).

$$D = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{\partial \ddot{x}}{\partial x} & \frac{\partial \ddot{x}}{\partial y} & \frac{\partial \ddot{x}}{\partial z} & \frac{\partial \ddot{x}}{\partial \dot{x}} & \frac{\partial \ddot{x}}{\partial \dot{y}} & \frac{\partial \ddot{x}}{\partial \dot{z}} \\ \frac{\partial \ddot{y}}{\partial x} & \frac{\partial \ddot{y}}{\partial y} & \frac{\partial \ddot{y}}{\partial z} & \frac{\partial \ddot{y}}{\partial \dot{x}} & \frac{\partial \ddot{y}}{\partial \dot{y}} & \frac{\partial \ddot{y}}{\partial \dot{z}} \\ \frac{\partial \ddot{z}}{\partial x} & \frac{\partial \ddot{z}}{\partial y} & \frac{\partial \ddot{z}}{\partial z} & \frac{\partial \ddot{z}}{\partial \dot{x}} & \frac{\partial \ddot{z}}{\partial \dot{y}} & \frac{\partial \ddot{z}}{\partial \dot{z}} \end{pmatrix} \tag{2.14}$$

2.4.2 Non-Autonomous system

Consider the state of a non-autonomous system represented by the vector X such that,

$$X = (t, x, y, z, \dot{x}, \dot{y}, \dot{z})^T \tag{2.15}$$

then the differential of the X with respect of time is given by,

$$\dot{X} = (1, \dot{x}, \dot{y}, \dot{z}, \ddot{x}, \ddot{y}, \ddot{z})^T \tag{2.16}$$

as,

$$\dot{t} = 1 \tag{2.17}$$

This is a way to include the dependency on time without increasing the complexity of the system. For this case, the variational matrix D_{NA} is given by Equation (2.18).

In Equation (2.18) the matrix D is the same as given by Equation (2.14).

$$D_{NA} = \left(\begin{array}{c|cccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & & & & & & \\ 0 & & & & & & \\ 0 & & & & & & \\ \frac{\partial \ddot{x}}{\partial t} & & & D & & & \\ \frac{\partial \ddot{y}}{\partial t} & & & & & & \\ \frac{\partial \ddot{z}}{\partial t} & & & & & & \end{array}\right) \tag{2.18}$$

2.5 Variational Equations

Suppose we have a system with state given by,

$$X = (t, X_1, X_2, \cdots, X_6)^T \tag{2.19}$$

such that,

$$\dot{X} = \begin{pmatrix} 1 \\ X_4 \\ X_5 \\ X_6 \\ \ddot{X}_1 \\ \ddot{X}_2 \\ \ddot{X}_3 \end{pmatrix} \tag{2.20}$$

Now the idea of implementation of 7×7 variational equations will be elaborated. Out of these 49 equations, one is the Equation (2.17) and six of them are given by Equation (2.20). Rest of the equations are given as [6],

$$\frac{d}{dt}\begin{pmatrix} X_7 \\ X_8 \\ X_9 \\ \vdots \\ X_{12} \end{pmatrix} = D \begin{pmatrix} X_7 \\ X_8 \\ X_9 \\ \vdots \\ X_{12} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{\partial \dot{X}_4}{\partial t} \\ \frac{\partial \dot{X}_5}{\partial t} \\ \frac{\partial \dot{X}_6}{\partial t} \end{pmatrix} \tag{2.21}$$

where the matrix *D* is given by Equation (2.14). Further,

$$\frac{d}{dt}\begin{pmatrix} X_{13} & X_{19} & \cdots & X_{43} \\ X_{14} & X_{20} & \cdots & X_{44} \\ \vdots & \vdots & & \vdots \\ X_{18} & X_{24} & \cdots & X_{48} \end{pmatrix} = D \begin{pmatrix} X_{13} & X_{19} & \cdots & X_{43} \\ X_{14} & X_{20} & \cdots & X_{44} \\ \vdots & \vdots & & \vdots \\ X_{18} & X_{24} & \cdots & X_{48} \end{pmatrix} \tag{2.22}$$

again, the matrix D is given by Equation (2.14). Equations (2.21) and (2.22) together are termed as the variational equations.

Now suppose we have an initial point of the trajectory of a spacecraft and we want to know what will be the change in the final position if the initial conditions are changed. To compute this, first, we start with a state transition matrix at the initial point which will of course be the identity matrix I as stated in Section 2.2. At the initial point (of the orbit, for example) we define the values of X_7 to X_{12} in Equation (2.21) as,

$$\begin{pmatrix} X_7 \\ X_8 \\ X_9 \\ \vdots \\ X_{12} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{2.23}$$

We initialise the matrix given on the right side of Equation (2.22) as,

$$\begin{pmatrix} X_{13} & X_{19} & \cdots & X_{43} \\ X_{14} & X_{20} & \cdots & X_{44} \\ \vdots & \vdots & & \vdots \\ X_{18} & X_{24} & \cdots & X_{48} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{2.24}$$

After the initialisation of these values we can integrate the trajectory of the spacecraft for a desired time. At the end of the computations, the State Transition Matrix can be obtained from the final values of the matrix composed of the elements from X_{13} to X_{48}. After we have the final State Transition Matrix we can estimate the final point if the initial conditions are changed with a small perturbation from Equation (2.3). This way allows to estimate the final point of a trajectory for different values of perturbations in the initial points. It allows for the reduction in terms of computation time as integrations need not be performed for different initial conditions. Also, the State Transition Matrix for a wide number of points on the initial trajectory can be stored and thus the whole new trajectory with perturbations can be estimated. Of course, this method will not give the exact trajectory but the accuracy can be increased by considering higher order terms in Equation (2.3).

2.6 Variational Matrix for Different Problems

2.6.1 N-Body Problem with respect to the Solar System Barycentre

This section will elaborate the formation of variational equation for the N-body problem w.r.t the solar system barycentre which was discussed in Equation (1.39). The state vector for the N-body problem is given as Equation (2.15) and its derivative is represented by Equation (2.16). In case of this particular problem,

$$\ddot{x} = (\ddot{\vec{r_i}})_x \quad \textbf{(2.25)}$$

$$\ddot{y} = (\ddot{\vec{r_i}})_y \quad \textbf{(2.26)}$$

$$\ddot{z} = (\ddot{\vec{r_i}})_z \quad \textbf{(2.27)}$$

where the values of $(\ddot{\vec{r_i}})_x$, $(\ddot{\vec{r_i}})_y$, and $(\ddot{\vec{r_i}})_z$ are given by equations (B.2), (B.3), and (B.4) respectively.

The variational matrix for this problem is constructed using the Equation (2.18). For this case, the values of $\frac{\partial \ddot{x}}{\partial t}$, $\frac{\partial \ddot{y}}{\partial t}$, and $\frac{\partial \ddot{z}}{\partial t}$ are taken from Equations (B.14), (B.15), and (B.16) respectively. For the calculation of matrix *D* from equation (2.14) the values can be taken from Equation (B.5) till Equation (B.13). It is important to note that for this case,

$$\frac{\partial \ddot{x}}{\partial \dot{x}} = \frac{\partial \ddot{x}}{\partial \dot{y}} = \frac{\partial \ddot{x}}{\partial \dot{z}} = \frac{\partial \ddot{y}}{\partial \dot{x}} = \frac{\partial \ddot{y}}{\partial \dot{y}} = \frac{\partial \ddot{y}}{\partial \dot{z}} = \frac{\partial \ddot{z}}{\partial \dot{x}} = \frac{\partial \ddot{z}}{\partial \dot{y}} = \frac{\partial \ddot{z}}{\partial \dot{z}} = 0 \quad \textbf{(2.28)}$$

After the matrix *D* has been created, it is can be used to compute the variational equations from the Equations (2.21) and (2.22). Further treatment is dealt with in Appendix B.

2.6.2 N-Body Problem with respect to a body in Solar System

This section will elaborate the formation of variational equation for the N-body problem w.r.t a body in the solar system which was evaluated in Equation (1.45). The treatment is similar to the previous sub-section and in this case too the state vector is given by Equation (2.15) and its derivative is represented by Equation (2.16). Again,

$$\ddot{x} = (\ddot{\vec{r_i}})_x \quad \textbf{(2.29)}$$

$$\ddot{y} = (\ddot{\vec{r_i}})_y \quad \textbf{(2.30)}$$

$$\ddot{z} = (\ddot{\vec{r_i}})_z \quad \textbf{(2.31)}$$

where the values of $(\ddot{\vec{r_i}})_x$, $(\ddot{\vec{r_i}})_y$, and $(\ddot{\vec{r_i}})_z$ are given by equations (C.7), (C.8), and (C.9) respectively.

The variational matrix for this problem is constructed using the Equation (2.18). For this case, the values of $\frac{\partial \ddot{x}}{\partial t}$, $\frac{\partial \ddot{y}}{\partial t}$, and $\frac{\partial \ddot{z}}{\partial t}$ are taken from Equations (C.19), (C.20), and (C.21) respectively. For the calculation of matrix *D* from equation (2.14) the values can be taken from Equation (C.10) till Equation (C.18). It is important to note that for this case also,

$$\frac{\partial \ddot{x}}{\partial \dot{x}} = \frac{\partial \ddot{x}}{\partial \dot{y}} = \frac{\partial \ddot{x}}{\partial \dot{z}} = \frac{\partial \ddot{y}}{\partial \dot{x}} = \frac{\partial \ddot{y}}{\partial \dot{y}} = \frac{\partial \ddot{y}}{\partial \dot{z}} = \frac{\partial \ddot{z}}{\partial \dot{x}} = \frac{\partial \ddot{z}}{\partial \dot{y}} = \frac{\partial \ddot{z}}{\partial \dot{z}} = 0 \quad \textbf{(2.32)}$$

After the matrix *D* has been created, it can be used to compute the variational equations from the Equations (2.21) and (2.22). Further treatment is dealt with in Appendix

C.

2.6.3 Quasi-Periodic Formulation of the Restricted N-Body Problem

This section will elaborate the formation of variational equation for the quasi-periodic formulation of the restricted n-body problem which were detailed in Section 1.8. The state vector is given by equation (2.15) and its derivative is represented by (2.16). The values of $\ddot{x}$, $\ddot{y}$, and $\ddot{z}$ are given by equations (1.141), (1.142), and (1.143) respectively.

The variational matrix for this problem is constructed using the Equation (2.18). For this case, the values of $\frac{\partial \ddot{x}}{\partial t}$, $\frac{\partial \ddot{y}}{\partial t}$, and $\frac{\partial \ddot{z}}{\partial t}$ are taken from Equations (D.22) to Equation (D.24) respectively.

For the calculation of matrix D from equation (2.14) the values can be taken from the Annexure D. After the matrix D has been created it is can be used to compute the variational equations from the Equations (2.21) and (2.22).

Chapter 3

PACKAGE DESCRIPTION

3.1 Introduction

The aim of this chapter is to introduce the package that was designed as a part of this thesis. It describes some important routines and other important aspects of the program. The purpose of this chapter is serve as a guide to every user who plans to implement routines in this package and to use the package to perform integrations.

3.2 The Package

3.2.1 Directory Structure

The Directory structure of the package is shown in Figure 3.1. There are two primary directories: *main* and *parallelshooting*. The directory *main* contains the JPL ephemeris in the folder *'/main/ephemerides/'*. The folder *'/main/spicelib/'* holds the JPL SpiceLib. The folder *'/main/source/'* contains the routines written in FORTRAN 77 with a *makefile*. When these routines are compiled they are copied and stored in the folder *'/main/lib/'*. The folder *'/main/examples/'* contains all the examples and test routines that were designed to test certain aspects of the package. These tests are described in the next chapter.

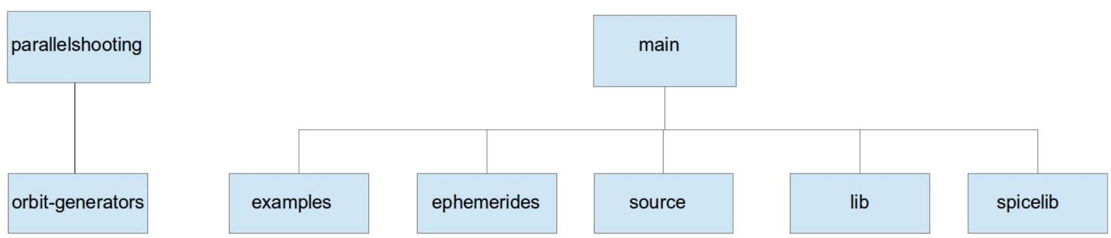


Figure 3.1 Package Directory Structure

3.2.2 Installation

Before the use of this package it must be compiled on the user's machine. To do so enter the directory *'/main/'* and execute the file *install.sh*. This script will compile the SpiceLib and the program library. After the installation is complete the user can implement routines in this package. It is very important to note that this package is intended for a 64-bit machine as the SpiceLib is 64 bit. The 32 bit SpiceLib has also been provided in the folder *'/main/spicelib32/'*. If the user intends to use the package on a 32-bit machine then the user must execute the script named *'./main/install32.sh/'* rather than *'./main/install.sh/'*. It is recommended to use the 64 bit version. After the program has been installed, the user can copy the file '*./main/lib/libjpl.a*' to any folder. While writing new routines the user can compile this library with their routines.

3.2.3 Using the Package

After the installation and compilation has been completed the user can build new routines for using the package. It is important that each new routine(for integrations or other purposes) must be accompanied by a copy of the file '*modeleph.dat*', which has been described in the Section 3.4.1. This file can be copied from the directory '*/main/source/*'.

3.3 SpiceLib

3.3.1 Ephemeris

Ephemeris or Ephemerides describe the position and velocity of a celestial object as a function of time. Ephemeris can be either analytical or numerical. Analytical ephemeris expresses the positions and velocities of an object in terms of algebraic equations of motion. Sometimes, the equations of motion of a body can be quite complicated and it can be extremely time consuming to find an analytical solution. Numerical Ephemeris, on the other hand, expresses the positions and velocities based on a table representing the numerical solution of the equations of motion to a certain degree of accuracy. A drawback of numerical ephemeris is that sometimes the size of tables could be very long. Thus, to reduce the size, sometimes the numerical ephemeris can be approximated by analytical functions. The ephemeris that will be used by this thesis is a numerical ephemeris by JPL DE406. It contains a total of 52 bodies. The bodies are shown in Appendix G. This ephemeris stores data from 3000 BC to 3000 AD [7] [9] and are fitted with Chebychev polynomials. To read this ephemeris, JPL has a special set of programs called as SpiceLib.

3.3.2 The SpiceLib

SpiceLib is a set of programs by NASA's Jet Propulsion Laboratory and has the capability to read the numerical ephemeris. Detailed discussion of this library is out of scope of this thesis. More information can be found in the reference [7] and [8]. The package designed as part of this thesis has the ability to access the SpiceLib to get data from the ephemeris.

3.4 Source Code Description

3.4.1 System Model

This section describes a very important file, namely, 'modeleph.dat'. This file is contained in the directory '*/main/source/*' and is used to initialise the model for performing numerical computations. It can be used to select bodies in the solar system that have to be considered for computations. An example model file has been shown in Annexure F. The file has four columns. The first column indicates the identifier of the body. The identifier of the body is an integer. The identifiers of the barycentre of all the systems are shown in Table 3.1. The bodies in a system are identified as follows. Suppose that the identifier of a barycentre is *P*, then the identifier of the planet of the system will be *P99* and the identifiers of the moons start from *P01* to *P98*. For example, the identifier of Mars barycentre is *4*, then Mars would be *499* and the identifiers of the moons Phobos and Deimos will be *401* and *402* respectively.

The second column determines whether the body is being considered in the model. If the value is '1' then the body is considered and a '0' signifies that it is not being considered. If a body is not considered, the user will not be able to access its information from the ephemeris and the gravitational contributions of this body will not be taken into account for the vector fields. The user of the package must open this file and select the body according to the model requirements. The third column gives the reference name of the ephemeris where the body is stored in the SpiceLib. The fourth column exists only for the barycentres and stores the gravity model status of the system . Gravity model has been described in the Section 3.4.3.

3.4.2 Common Model

FORTRAN 77 has no global variables. Thus, as soon as the routine finishes execution the variables are lost. To pass variables between routines using a common model is one of the options available in FORTRAN. These common variables can be defined in every routine or can be passed in through a file. This software package uses the latter. A file named 'commonmodel.inc' is used to pass these common variables. The variables passed are discussed below. This file has been stored in the directory '*/main/source/*'. Although all of the common variables are important and are used by nearly all the source files, some of the common variables are more important as these must be clearly understood by the user. Thus, for the sake of description, they are divided into two categories: *Important* and *Internal*.

Important Common Variables

- **ICO:** This variable is used to indicate the ID of the secondary body in the restricted three body system. For example, if we are considering Sun-Planet in an RTBP like system then *ICO* will hold the ID of the planet. In case of Sun-Jupiter system the value of ICO will be 599. On the other hand if we are considering Planet-Satellite, such as Jupiter-Io system, ICO will store the ID of the satellite, in afore-mentioned case ICO will be 501.

- **IBC:** IBC holds the origin of the coordinate system. IBC can hold ID of any body in the ephemeris. If IBC is set to zero, the coordinates are taken with respect to the solar system barycentre.

- **LIST(3,IMAX):** It is a array of three rows and IMAX columns and it contains information about the model considered. The first row of LIST contains the ID of the body. The second row shows if we are considering position of the body (takes

Table 3.1 Identifier of the Bodies in Ephemeris Model

Body	Identifier
Sun	10
Mercury Barycentre	1
Venus Barycentre	2
Earth Barycentre	3
Mars Barycentre	4
Jupiter Barycentre	5
Saturn Barycentre	6
Uranus Barycentre	7
Neptune Barycentre	8
Pluto Barycentre	9

the value one); position and velocity (value two); position, velocity and acceleration (value three); or, position, velocity, acceleration and over- acceleration (value four). And the third row indicates if is a barycentre (value zero), a planet (value one) or a satellite (value two). Since it has 53 columns it can hold if all the bodies in the ephemeris have been selected. In general, if bodies less than 53 have been selected, rest of the variable will be set to zero.

- **SECONDARYINDEX:** This hold the column index of the secondary body in the array LIST.

- **PRIMARYINDEX:** This hold the column index of the primary body in the array LIST.

Internal Common Variables

- **IMAX**: This variable is used to define the size of other common variables. Since the software package is capable of working with maximum of 53 bodies, IMAX has been defined as 53.

- **SEMIAXFIN:** It is a vector of a length IMAX. It holds the value of semi-major axis of the bodies. If the body is a planet, the value is the semi-major axis of the orbit of the planet around the planet. If the body is a satellite, the value is the semi-major axis of the orbit of the satellite around the its planet. For example, the semi-major axis of Io will be the value around Jupiter. The semi-major axis of a barycentre is same as the planet i.e. Jupiter Barycentre will have the same value as Jupiter.

- **SEMIINDEX:** It is a vector of a length IMAX. It holds the value of ID of the bodies. The point of having this vector is that the index of the body in this vector will be the same as the index in SEMIAXFIN.

- **XMU:** It stores is the mass parameter of the restricted three body system considered.

- **ENEM:** It stores the mean motion in day^{-1} of a selected system. It has been computed using Kepler's third law by the routine **INTXMUENEM**.

- **BARYINDEX:** This hold the column index of the barycentre in the array LIST. It is important to note that if the system considered is the Sun-Planet Barycenter then BARYINDEX is 0.

- **C1, C2, C3:** C1, C2, and C3 are three column vectors of length 3. They store the value for the columns of the matrix C necessary for coordinate transformations. It is important to note that the these variables are only initialised if the routine **CALCPARAMRTBP** has been called. Otherwise they are empty. These are computed from Equations (1.11), (1.12), and (1.13) respectively.

- **C1P, C2P, C3P:** C1P, C2P, and C3P are three column vectors of length three. They store the value for the columns of the matrix C' necessary for coordinate transformations. These are calculated from Equations (1.22), (1.24), and (1.23) respectively.

- **C1PP, C2PP, C3PP:** C1PP, C2PP, and C3PP are three column vectors of length 3. They store the value for the columns of the matrix C'' necessary for coordinate transformations. Their values are obtained from Equations (1.33), (1.33), and (1.33) respectively.

- **C1PPP, C2PPP, C3PPP:** C1PPP, C2PPP, and C3PPP are three column vectors of length 3. They store the value for the columns of the matrix C''' necessary for coordinate transformations. These are calculated from Equations (E.2), (E.8), and (E.3) respectively.

- **XT:** This variable stores the value of the vector $\vec{b}$ required for the transformations as described in Chapter 1. The value is computed from the Equation (1.17).

- **XK, XKP, XKPP, XKPPP:** The variables XK, XKP, XKPP, and XKPPP store the value of the constants k, k', k'', and k''', respectively, used for coordinate transformations. Their values are obtained from Equations (1.16), (1.25), (1.37), and (E.7).

- **ADLIST:** This variable is a matrix with IMAX columns and three rows. This variable stores the backup of the the variable LIST in case the routine **ADAPTLIST** is used. List adaptation has been described in section 3.4.4.

- **ADFLAG:** This variable can take up two values 1 and 0 depending on whether adapted list is being used or not. The value of this variable is initialised by **ADAPTLIST** and must only be used as a read only value.

- **INDEXBACK:** This column vector of length three stores the indexes of the primary, secondary, and the barycenter when the routine **ADAPTLIST** is being used.

- **GRAVLIST:** It is an array of length 9 and has two columns. The first column contains the ID of the nine barycentres of the nine planets. The second contains the status of gravity model. Gravity model has been detailed in Section 3.4.3

3.4.3 Gravity Model

Suppose that integrations are being performed in a vector field for a particular model of the solar system. If, for example, the body under consideration (fro example, a spacecraft) is in vicinity of Jupiter system i.e. Jupiter and its satellites then it is more accurate to consider every body in this system. But, if our body is very far away from the Jupiter satellite systems then it makes more sense to consider the Jupiter System as one single point system based on the centre of gravity of the system. This can reduce the amount of numerical computations as the program will not calculate vectors and coordinate transformations for each of the Jovian satellite but will only compute such factors for one body. To facilitate this, Gravity model has been suggested. The gravity model is elaborated in the Table 3.2.

3.4.4 List Adaptation

List adaptions can be useful in case the user wants to perform integrations just in a particular three body system and to remove all the perturbations of the other bodies in the model. To facilitate this, the package has a routine called **ADAPTLIST**. If this routine is used it removes every body from the list but the primary, secondary, and the barycenter defined in the initial model.

3.5 Subroutines

This section will describe the important subroutines that are included in this software package and are being used in course of this study. Some of the routines we will define as ‘internal’, which means that the user need not know about it and are only used by some other routines, but are important enough to be mentioned. The routines are grouped into categories.

Table 3.2 Gravity Model Indicators

Gravity Model Status	Bodies Considered
0	Barycentres Only
1	Only the planet
2	Only the satellites
3	Planet and satellites

3.5.1 Basic System and Model Definition

These routines are the basis routines that are used to setup the computations and used to initialise the systems. Some of them are important and the user must be aware of them.

READ_MODELEPH

The main purpose of this routine is to read the file '*modeleph.dat*', from the directory in which the user is implementing routines, and to write the common variable *LIST*. The first step is to Figure 3.2 shows the flowchart of the working of the routine. The first step is to open the file '*modeleph.dat*' and to read its columns. The next step is to check if the value of *ICO* has been defined and is not equal to zero. In case *ICO* is zero, the program stops. Otherwise the routine **CHANGE_FILE_DAT** is called This routine writes the selected bodies into a file named '*new_files.dat*'.
The next step is to call the routine **CONG**. This routine verifies the integrity of the model i.e. if a planet has been selected then its barycenter must also be selected. If a satellite has been selected, it is mandatory to select its planet and the planet's barycenter. After these checks are made the routine **READ_MODELEPH** will write these selected bodies into *LIST*. This is an internal routine.

SUBROUTINE NMODJPL(P1, P2, P3)

1. INTEGER P1: The identifier of the body which will serve as the origin of the reference system.
2. INTEGER P2: The value of *ICO* i.e. The identifier of the body to serve as the secondary in the restricted three body system. If the input is zero then no RTBP system is initialised.
3. INTEGER P3: The value of the Libration point (1 to 5) around which the integrations are intended to be performed. Right now this feature has not been implemented.

This routine is responsible to initialise most of the common variable and is the first routine that must be called. For example, if we want to work with Jupiter-Io system, around the libration point number two, and we want to perform integrations with respect to the solar system barycenter then the call to the routine will be as follows,

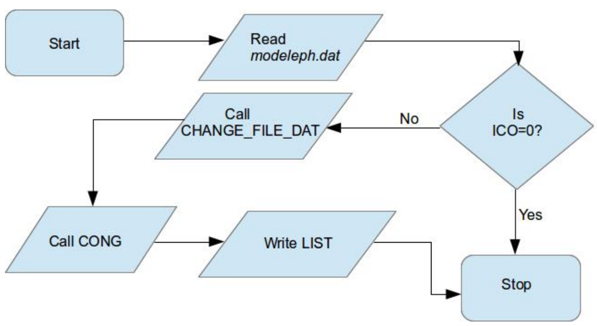


Figure 3.2 READ_MODELEPH Flowchart

CALL NMODJPL(0,501,2)

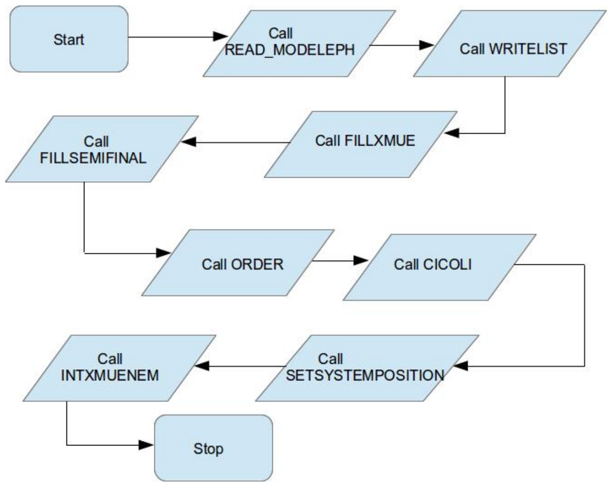


Figure 3.3 NMODJPL Flowchart

The flowchart of working of the routine has been shown in Figure 3.3.The routine **NMODJPL** calls the routine **READ_MODELEPH** followed by a call to **WRITELIST**. The routine **WRITELIST** writes a file named '*list.dat*'. The next step is to call the routine **FILLXMUE** which returns a vector containing the gravitational constants i.e. *GM* of all the bodies in the ephemeris. The routine **FILLSEMIFINAL** is then called which initialises the common variables *SEMIAXFIN* and *SEMIINDEX*. The next step is a call to the routine **ORDER** which makes sure that the variable *LIST* is ordered. The routine **CICOLI** is eventually called which is responsible for initialisation of the values of *ICO* and *LI*. The routine **SETSYSTEMPOSITIONS** sets the values for the common variables *PRIMARYINDEX*, *SECONDARYINDEX*, and *BARYINDEX*. Afterwards the routine **INTXMUENEM** is called which is responsible for calculation of the values of common variables *XMU* and *ENEM*.

SUBROUTINE PLAJPLNSAT(P1, P2, P3)

1. REAL*8 P1: Date in the equatorial J2000 reference frame for which the states are required.

2. INTEGER P2: IPVA which can take integer values from '1' to '4'. If the value of IPVA is '1' then the routine only returns positions of the bodies. A value of '2' returns positions and velocities, '3' returns positions, velocities, and accelerations, and '4' returns positions, velocities, accelerations, and over-accelerations.

3. REAL*8 P3(12, IMAX) : Matrix to hold the output of the routine.

This routine is responsible for returning states of bodies included in the model. The routine calls the routine **SPKG** which reads the SpiceLib and returns the vector with positions, velocities, accelerations, and overaccerlerations. The coordinates of barycentres are with respect to the solar system barycenter. The coordinates of planets and moons are with respect to the barycenter of the system. For example, the coordinates of Jupiter's moon Europa will be given with respect to the Jupiter Barycenter. The routine then calls **PVASIBC** which returns the coordinates of all the bodies with respect to the *IBC* i.e. the centre of the coordinate system selected by the user. The transformations are carried using the formulae given in Section 1.3.2. So finally the output matrix is a matrix of size (12, IMAX), where IMAX is 53 (see Section 3.4.2). The 12 rows will hold state of the body. It will contain the requested values (according to IPVA) with respect to the user selected reference frame. The columns will be in the same order as that of the variable *LIST*.

For example, a call to the routine will look as,

CALL PLAJPLNSAT(1000.D0,4,PVAS)

where, PVAS should be a 12 by 53 matrix to hold the output produced by the routine.

SUBROUTINE SEARCHLIST(P1,P2)

1. INTEGER P1: The identifier of the body for which the index is needed.
2. INTEGER P2: Variable to store the output.

The routine searches the common variable *LIST* and returns the index of the position of the requested body. The routine is useful as it saves time for the user because it prevents writing loops to search the list manually. Of course, it is necessary that the requested body must have been selected in the model.

The importance of this routine can be seen from the following example. If the routine is called as,

CALL SEARCHLIST(502,POS)

then the routine will return the index of the Europa in the variable *POS*. If now for example we need the position of Europa then the position will be *[PVAS(1,POS), PVAS(2, POS), PVAS(3,POS)]*.

SUBROUTINE RETURNSEMIMAJAXIS(P1,P2)

1. INTEGER P1: the identifier of the body for which the semi-major axis is needed.
2. REAL P2: Variable to receive the semi-major axis.

The routine returns the semi-major axis of the requested body. The values of semi major axis have been taken from *The Explanatory Supplement to the Astronomical Almanac* (see [9]). The routine searches the common variable *SEMIINDEX* for the index of the body and uses this index to return the semi-major axis from the common variable *SEMIAXFIN*. The routine is called as,

CALL RETURNSEMIMAJAXIS(301,SMA)

In this example the routine will return the semi-major axis of the Moon in the variable *SMA*.

SUBROUTINE RETURNXMUEA(P1, P2)

1. INTEGER P1: The identifier of the body for which the value is needed.
2. REAL*8 P2: Variable to receive the output of the subroutine.

The routine returns the value of GM of the requested body, where, *G* is the Gravitational Constant and *M* is the mass of the body. The values have been taken from *The Explanatory Supplement to the Astronomical Almanac* (see [9]). The routine searches the common variable *SEMIINDEX* for the index of the body and uses this index to return the value of GM from the common variable *XMUE*. The routine is called as,

CALL RETURNXMUEA(301,XMUEM)

In this example the routine will return the value of GM of the Moon in the variable *XMUEM*.

3.5.2 Coordinate Transformations

These routines can be used for the transformation from one reference frame to the other. These are very important as many tests and computations depend on these routines.

SUBROUTINE CALCPARAMRTBP(P1,P2)

1. REAL*8 P1: The time in equatorial J2000 reference frame
2. INTEGER P2: The value of IPVA. IPVA can take integer values from '1' to '4'. For example, if during coordinate transformations only positions and velocities must be transformed then IPVA must be '2'.

This internal routine calculates the parameters required for implementation of the coordinate transformations. It calculates the values of the common variables *C1*, *C2*, *C3*, *C1P*, *C2P*, *C3P*, *C1PP*, *C2PP*, *C3PP*, *C1PPP*, *C2PPP*, *C3PPP*, *XT*, *XK*, *XKP*, *XKPP*, and *XKPPP*. The call to this routine is,

CALL CALCPARAMRTBP(1000.D0, 2)

The routine uses formulae given in Section 1.3.1.

SUBROUTINE TRANS(P1,P2,P3,P4,P5)

1. REAL*8 P1: The time in equatorial J2000 reference frame.
2. REAL*8 P2(12): Equatorial coordinates in form of a one dimensional array.
3. REAL*8 P3(12): Adimensional coordinates in form of a one dimensional array,
4. INTEGER P4: IPVA, takes values from '1' to '4'.

5. INTEGER P5: The request index and it can take two values: ‘1’ and ‘-1’. If the request index is ‘1’ then equatorial coordinates are transformed into adimensional ones. On the other hand, if it is ‘-1’ then adimensional coordinates are transformed to equatorial coordinates.

This routine can be used to transform from equatorial coordinates to adimensional RTBP coordinates or vice-versa. The routine uses the formulae given in Section 1.3.1 for the conversion.

For example, if we want to transform equatorial velocities into adimensional velocities on the day ‘1000’ then a typical call to this routine will be like,

CALL TRANS(1000.D0,EQ,AD,2,1)

In this example, *EQ* must be an array of length ‘6’ (3 for positions, 3 for velocities) and must contain the coordinates that need to be transformed. Since we need adimensional coordinates, in this case the *AD* must be an empty array of length ‘6’.

SUBROUTINE TRANSIBC(P1,P2,P3,P4,P5,P6)

1. REAL*8 P1: The time in equatorial J2000 reference frame.
2. REAL*8 P2(12): The input containing points to be transformed (let’s call it X_{old})
3. INTEGER P3: The identifier of the origin of the old reference frame with respect to which the coordinates X_{old} are specified.
4. REAL*8 P4(12): An empty one dimensional array vector to store the output of the routine (let’s call it X_{new})
5. INTEGER P5: The identifier of the body of the new origin of reference frame with respect to which the coordinates X_{new} are specified
6. INTEGER P6: Value to specify what is transformed. If it is ‘1’ only positions are transformed, ‘2’ transforms positions and velocities, ‘3’ transforms positions, velocities, and accelerations, ‘4’ transforms positions, velocities, accelerations, and over-accelerations.

This routine can be used to transform from equatorial coordinates with respect to one origin of reference frame to equatorial coordinates with respect to another origin of reference frame. It should be noted that the length of the arrays X_{old} and X_{new} must be at least ‘3*IPVA’ to properly store the output of the routine.

An example call to the function will look like,

CALL TRANSIBC(1000.D0,XOLD,10,XNEW,599,3)

In this example, the routine will transform positions, velocities, and accelerations (as IPVA is 3) stored in the variable *XOLD* from a reference frame centred in the Sun (with ID 10) to a new reference frame centred in Jupiter (with ID 599) and will store the output in the variable *XNEW*. Again, the length of *XNEW* and *XOLD* mus be at least 9 (3*IPVA). The routine uses the formulae from the Section 1.3.2.

SUBROUTINE CDJTA(P1,P2,P3,P4)

1. REAL*8 P1: The date in equatorial J2000 time.
2. REAL*8 P2: The adimensional time.
3. REAL*8 P3: Origin of the equatorial time frame. The origin of the equatorial time frame must be supplied always as '0.D0'. This option can allow user to use an equatorial reference frame with a different epoch than the date '0'.
4. INTEGER P4: The request index which takes two integer values: '1' and '-1'. If the routine is called with the value '1' then the equatorial time is transformed to adimensional time. The adimensional time is transformed to equatorial time if the request index is '-1'.

This routine can be used to transform from equatorial time to adimensional time or vice-versa. An example call to the routine will look like,

CALL CDJTA(1000.D0,TA,0.D0,1)

This call will transform the equatorial date '1000' to adimensional time and will store it in the variable *TA*. This routine uses concepts of transformation of time given in Section 1.3.1.

3.5.3 Vector Fields

These routines implement vector fields and the variational equations and can be used by the user to perform integrations.

SUBROUTINE VFSSB(P1,P2,P3,P4)

1. REAL*8 P1: The time in equatorial J2000 reference frame.
2. REAL*8 P2(48): The state where the vector field and variational equations must be calculated
3. INTEGER P3: Integer variable *N* specifying whether only the vector field is needed or the variational equations are needed as well. The variable *N* can have two values: '6' means only the vector field are returned, '48' means the vector field as well as the variational equations are computed and returned.
4. REAL*8 P3(48): Array to hold the output of the routine.

This routine computes the vector field and the variational equations for a body in the solar system with respect to the solar system barycentre depending on the model specified by the user. The equations for the vector field are given in Equation (1.39) and the variational equations are given in Section 2.6.1. An example of a call to the routine looks like,

CALL VFSSB(1000.D0, B, 48, F)

Here, the routine will compute the value of the variational equations and the vector fields on the date '1000' based on the positions supplied in array *B* and will return the output in the array *F*. It should be noted that the routine only works if the reference frame is centred in the solar-system barycenter.

SUBROUNTINE VFIBC(P1,P3,P4,P5)

1. REAL*8 P1: The time in equatorial J2000 reference frame.
2. REAL*8 P2(48): The state where the vector field and variational equations must be calculated
3. INTEGER P3: Integer variable *N* specifying whether only the vector field is needed or the variational equations are needed as well. The variable *N* can have two values: '6' means only the vector field are returned, '48' means the vector field as well as the variational equations are computed and returned.
4. REAL*8 P3(48): Array to hold the output of the routine.

This routine computes the vector field and the variational equations for a body in the solar system with respect to a body specified as the origin of the equatorial reference frame. The reference body must be stored in the common variable *IBC*. The equations for the vector field are given in Equation (1.45) and the variational equations are given in Section 2.6.2. The variable *N* can have two values: '6' means only the vector field are returned, '48' means the vector field as well as the variational equations are computed and returned.

An example of a call to the routine looks like,

CALL VFIBC(1000.D0, B, 48, F)

Here , the routine will compute the value of the variational equations and the vector fields on the date '1000' based on the positions supplied in array *B* and will return the output in the array *F*. It should be noted that the routine only works if the reference frame is centred in the some body other than the solar system barycenter i.e. $IBC \neq 0$.

3.5.4 Gravity Model and List Adaptation

SUBROUTINE RETURNBODYSTATUS(P1,P2)

1. INTEGER P1: The identifier of the body
2. INTEGER P2: An empty variable to hold the output.

The purpose of this routine is to return whether a particular body must be considered for perturbations or not. The routine reads the gravity model and returns an Integer value. A return of '1' means that the body has to be considered and a return of value '0' means that the body should not be considered. A sample call to the routine is,

CALL RETURNBODYSTATUS(501,STAT)

In this example, the routine is asked to return the status for IO. It will search the gravity list and if the value in the gravity list for Jupiter Barycenter is either '2' or '3' then the routine will return the variable *STAT* with a value of '1' else with a value '0'.

SUBROUTINE ADAPTLIST

The purpose of this routine is to change the common variable *LIST*. The routine takes the existing *LIST* and stores it in the common variable *ADLIST*. After the backup is made, the routine removes all the bodies from *LIST* and puts in only the primary, secondary, and the barycenter for the defined RTBP system. For example, if the selected RTBP system is Saturn-Titan then the new list will contain only Saturn, Titan, and the barycenter of the Saturn satellite system. After changing the *LIST* this routine makes a backup of the common variables *PRIMARYINDEX*, *SECONDARYINDEX*, and *BARYINDEX* in the common variable array *INDEXBACK*. These variables are then set to ‘1’, ‘2’, and ‘3’ respectively as that is the new position of these bodies in the new ”adapted” *LIST*. It also sets the common variable *ADFLAG* to ‘1’ signifying that *LIST* has been adapted. Also, whenever the routine is called it checks that whether the variable *ADFLAG* is ‘0’ or not because a value ‘1’ would signify that the routine has been used once in the past. If the value of this variable is ‘1’ then this routine restores the original *LIST* from the backup stored in the variable *ADLIST* and restores the variables *PRIMARYINDEX*, *SECONDARYINDEX*, and *BARYINDEX* from the backup stored in the array *INDEXBACK*. Then it sets the variable *ADFLAG* back to ‘0’. This routine takes in no inputs. A sample call to the routine is,

CALL ADAPTLIST

Also, it is important to note that if the *LIST* is adapted then the routine **PLAJPLNSAT** will print out a message reminding the user of the same.

3.5.5 Numerical Differentiation and Integrations

Numerical differentiation becomes important to verify the analytical differentiation of some functions and can also be employed if analytical expressions of differential of some function is not available. Suppose we have a function $f(x, y, .., t)$ The numerical partial differential of a function with respect to a variable, say x, can be expressed as,

$$\frac{\partial f(x, y, .., t)}{dx} = \frac{f(x+h, y, .., t) - f(x-h, y, .., t)}{2h} \tag{3.1}$$

The double derivatives can be expressed as,

$$\frac{\partial^2 f}{\partial x^2} = \frac{-f(x-h, y, .., t) + 2f(x, y, .., t) - f(x+h, y, .., t)}{h^2} \tag{3.2}$$

$$\begin{aligned}\frac{\partial^2 f}{\partial xy} = \frac{1}{4h_1h_2}\Big(&f(x+h_1, y+h_2, .., t) - f(x+h_1, y-h_2, .., t) \\ &-f(x-h_1, y+h_2, .., t) + f(x-h_1, y-h_2, .., t)\Big)\end{aligned} \tag{3.3}$$

SUBROUTINE CALCOOACC(P1,P2)

1. REAL*8 P1: The time in the equatorial J2000 frame
2. REAL*8 P2(3,IMAX): An empty array to hold the output.

This routines calculates the over-over acceleration of a body. Since the package does not have the ability to return analytical value of the over-over acceleration and the values of the three components are calculated using numerical differentiation. The over-over acceleration is calculated for all the bodies in the model.

A sample call to this routine will look something like,

CALL CALCOOACC(1000.D0, OOVERACC)

This will return the over-over accelerations of all the bodies in the model in on the date '1000'.

SUBROUTINE RK78(P1,P2,P3,P4,P5,P6,P7,P8,P9,P10,P11)

1. REAL*8 P1: Current value of independent variable, in our case, date in some reference frame
2. REAL*8 P2(48): Current value of dependent variable, in our case, the vector field
3. INTEGER P3: Length of the vector holding the dependent variable
4. REAL*8 P4: The time step to be used
5. REAL*8 P5: Minimum allowed value of the time step
6. REAL*8 P6: Maximum allowed value of the time step
7. REAL*8 P7: Tolerance e_1
8. REAL*8 P8(13,N): Auxiliary variable of size (13, N), where *N* is the number of equation
9. REAL*8 P9(N): Empty auxiliary vector of size N
10. REAL*8 P10(N): Empty auxiliary vector of size N, to return the value of the output of the routine for which the integrations has to be performed
11. EXTERNAL P11: Name of the routine for which the integrations must be performed i.e. the name of the vector field.

This routine can be used to perform integrations in a vector field using Runge-Kutta-Fehlberg method of the order 7 and 8. For details into RK78 method and its numerical equations the reader can see Reference [12]. It uses a total of 13 steps and computes two different guess of the next point. Then the difference between the estimation is computed and the L_1 norm is obtained. If this norm divided by the number of equations is less than a given tolerance e_1 times (1+0.01*L_1 norm at the point) then this estimation is returned as the next point. If this condition is not satisfied then a new value of the step size *h* is obtained and the computation is done again. After the computations are completed a prediction of the next step *h* is returned. The routine takes the following inputs, in order: A call to this routine will look like,

CALL RK78 (1000.D0,XJP,6,1.8D-2,1.D-4,1.D0,1.D-13,R,B,F,VFSSB)

In this example, the integration will be performed on the day ‘1000’ for the coordinates stored in the vector *XJP*. The size of vector *XJP* must be at least 6, as the number of equations will be 6. The initial time step that will be used will be 1.8×10^{-2}. The minimum value for the time step will be 1×10^{-4} and the maximum value is 1. The tolerance e_1 will be 1×10^{-13} and the integrations will be performed for the routine **VFSSB**.

3.6 Sample Program

This section will give an example of how to start the program to assist the user to implement some parts of the program. A simple example is to obtain the trajectory of Moon with respect to a centre of the earth. The sample program is shown below.

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C THIS EXAMPLE SHOWS A BASIC ROUTINE IMPLEMENTATION IN THE
C PACKAGE
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
PROGRAM MAIN
IMPLICIT NONE

CHARACTER*64 ARXOUT
INTEGER I,MOON
REAL*8 PVAS(12,53), T, XMOONI(6),TOLRK,HMIN,HMAX
REAL*8 TFINAL, H, B(48),F(48),R(13,48)

EXTERNAL VFIBC
C
C INITIALISING THE SYSTEM (EARTH-MOON)
C

CALL NMODJPL(399,301,0)

C
C THE IBC SELECTED IS 399.  THUS REFERENCE FRAME WILL BE CENTRED
C IN THE EARTH
C
C THE ICO IS 301.  WHICH MEANS THAT THE SECONDARY IS THE MOON
C (SEE MODELEPH.DAT IN APPENDIX F)
C
C
C SELECTING A RANDOM INITIAL TIME
C
T=232.D0
C

C
C CALL TO THE ROUTINE PLAJPLNSAT TO GET THE POSITIONS OF ALL BODIES
```

```
C OF THE SYSTEM
C

CALL PLAJPLNSAT(T,2,PVAS)

C
C PVAS NOW STORES THE POSITION OF ALL THE BODIES IN THE SYSTEM
C

CALL SEARCHLIST(301,MOON) !SEARCH THE POSITION OF THE MOON

C
C STORING THE CURRENT POSITION AS AN INITIAL GUESS OF THE INTEGRATION
C

DO I=1,6
XMOONI(I)=PVAS(I, MOON)
END DO

C
C PARAMETERS FOR RK78
C
TOLRK=1.D-13
HMIN=1.D-4
HMAX=1.D0
H=1.D-1

C
C INTEGRATIONS FOR 60 DAYS (2 REVOLUTION PERIODS APPROX)
C

TFINAL=T+60.D0

C
C FILE FOR STORING THE OUTPUT
C

CALL GETLUN(IUNIT)
NCD=IUNIT
ARXOUT='output.dat'

C
C START OF THE INTEGRATIONS
C

OPEN (NCD,FILE=ARXOUT)
WRITE (NCD,100) T,(XMOONI(I),I=1,6)
```

```
50 CALL RK78 (T,XMOON(I),6,H,HMIN,HMAX,TOLRK,R,B,F,VFIBC)

C
C WRITING OUTPUT OF RK78 TO FILE
C

WRITE (NCD,100) T,(XMOONI(I),I=1,6)

100 FORMAT (1X,7E24.16)

IF (T.LT.TFINAL) GOTO 50 !  LOOP TILL FINAL TIME

CLOSE(NCD)

CALL CLOSEEPH ! CLOSE ALL OPEN FILES

RETURN

END
```

Chapter 4

TESTS AND EXAMPLES

4.1 Introduction

After the package was implemented it is important to test different aspects of the programs to verify the integrity and robustness of the package. The purpose of this chapter is to elaborate these tests. The chapter also elaborates some examples that demonstrate the capability of the package by performing integrations in some vector fields.

4.2 Test of Routines

4.2.1 PLAJPLNSAT

The routine **PLAJPLNSAT** has to be checked to ensure that the output from ephemeris are correct. To ensure this we will employ the numerical differentiation elaborated in the Section 3.5.5. The steps followed are as follows,

1. A sample body was selected. Value of the three components of its velocity was then calculated numerically differentiating the positions obtained from the ephemeris as per Equation (4.1). The Δt was taken to be $1.15740741 \times 10^{-8}$ days which is equal to $0.01s$. Similarly, the process can be followed to obtain numerical values of accelerations by differentiating velocities obtained from the ephemeris and over-accelerations can be obtained by numerically differentiating the accelerations obtained from the ephemeris.

$$v_x = \frac{p_x(T + \Delta t) - p_x(T - \Delta t)}{2\Delta t} \tag{4.1}$$

   where $p_x(T)$ represents the position of the body at time *T*.

2. The routine **PLAJPLNSAT** was called with *IPVA* equal to '4' for both time $T + \Delta t$ and $T - \Delta t$. This will make the routine return positions, velocities, accelerations, and over-accelerations for both the times.

3. The numerical values of velocities, acceleration, and over-acceleration and their values obtained were compared to the values obtained from ephemeris. The absolute error and the relative error was computed.

4. The process was repeated for *T* starting from '100' days to '1350' days with steps of '50' days for three different bodies.

The code used is in the file *'/main/examples/TEST-DIFFP/main.f'*. The results of the test are shown in Table 4.1.

Table 4.1 Results of Numerical Differentiation of PLAJPLNSAT: Relative Errors

Body	Δt (Days)	**Velocities**	**Accelerations**	**Over -Accelerations**
Jupiter	$1.15740741 \times 10^{-1}$	Mean of Components: 6.74474×10^{-6} Standard Deviation of Components: 2.30361×10^{-5}	Mean of Components: 8.48638×10^{-3} Standard Deviation of Components: 1.61496×10^{-2}	Mean of Components: 1.81640×10^{-2} Standard Deviation of Components: 0.14703
Io	Velocities and Accelerations: $1.15740741 \times 10^{-8}$ Over -Accelerations: $1.15740741 \times 10^{-1}$	Mean of Components: 8.15694×10^{-6} Standard Deviation of Components: 9.49089×10^{-6}	Mean of Components: 8.44216×10^{-6} Standard Deviation of Components: 1.41249×10^{-2}	Mean of Components: 9.48154×10^{-2} Standard Deviation of Components: 0.45644
Mars	Velocities and Accelerations: 6.09534×10^{-6} Over -Accelerations: $1.15740741 \times 10^{-1}$	Mean of Components: 6.81394×10^{-6} Standard Deviation of Components: 5.37963×10^{-6}	Mean of Components: 3.50977×10^{-7} Standard Deviation of Components: 6.63660×10^{-6}	Mean of Components: 9.89109×10^{-3} Standard Deviation of Components: 2.48941×10^{-2}

4.2.2 TRANS

The routine TRANS was tested to ensure that that the output of this routine is robust as this routine is used by a lot of other parts of the package. The steps to test were as follows:

1. Three different bodies were selected. The RTBP system was selected as Jupiter-Io system. The three bodies were Jupiter, Io, and Earth.
2. The equatorial coordinates (positions, velocities, and accelerations) of these bodies were stored. These were then transformed into RTBP coordinates.
3. The adimensional coordinates were then transformed back to equatorial and relative error was calculated.

The result of the test are shown in Table 4.2. The code used is in the file '*/main/examples /TEST-JPLTOADIM/main.f*'.

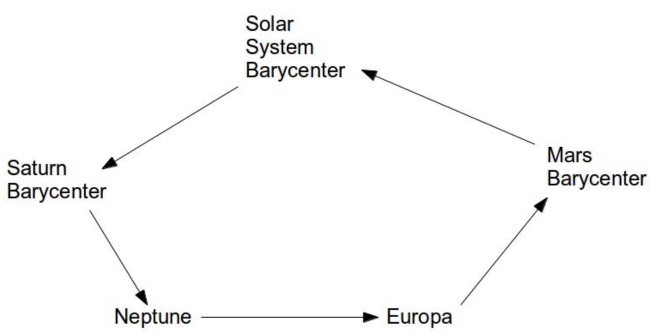
Table 4.2 Relative Errors for TRANS: Date 1002.D0

Body	Positions	Velocities	Accelerations
Jupiter	x=0.00000 y=0.00000 z=0.00000	x=0.00000 y=0.00000 z=0.00000	x=1.22820×10^{-16} y=-2.08547×10^{-16} z=1.18221×10^{-16}
IO	x=0.00000 y=0.00000 z=0.00000	x=0.00000 y=1.31404×10^{-15} z=4.91977×10^{-16}	x=1.43098×10^{-16} y=2.43422×10^{-16} z=3.73110×10^{-16}
Earth	x=5.99058×10^{-16} y=-9.57779×10^{-15} z=-1.25438×10^{-14}	x=-2.83676×10^{-12} y=-3.40266×10^{-13} z=-9.39320×10^{-14}	x=-3.70144×10^{-12} y=-7.70256×10^{-10} z=-1.62670×10^{-9}

4.2.3 TRANSIBC

The idea of testing this routine is that a set of coordinates are transformed from one reference system to several others and finally they are transformed back to the initial reference frame. The reference systems used are shown in Figure 4.1.

Figure 4.1 TRANSIBC Reference Frame Transformation



The transformation was carried on the date ‘110’. The states of Venus with respect to the Solar System barycenter at this date were stored and transformed into the reference frames given in Figure 4.1. The results are shown in Table 4.3. The code used is in the file '*/main/examples/TEST-TRANSIBC/main_test.f*'.

Table 4.3 Maximum Relative Errors for TRANSIBC

Body	Positions	Velocities	Accelerations	Over-Accelera--tions
Venus	x=-2.08106×10^{-15}	x=0.00000	x=-6.96743×10^{-16}	x=-3.46464×10^{-13}
	y=7.67402×10^{-14}	y=0.00000	y=-3.58038×10^{-14}	y=1.28933×10^{-14}
	z=7.58649×10^{-15}	z=0.00000	z=9.10535×10^{-15}	z=-1.47874×10^{-13}
Io	x=-2.31526×10^{-16}	x=3.60894×10^{-16}	x=-1.84372×10^{-16}	x=-1.45090×10^{-16}
	y=-7.17070×10^{-16}	y=0.00000	y=-1.40284×10^{-16}	y=1.14290×10^{-16}
	z=-2.96177×10^{-16}	z=0.00000	z=1.64900×10^{-16}	z=0.00000

4.3 Integrations

One of the primary applications of the package is to perform numerical integrations in the solar system. The integrations are performed using the following basic steps:

1. A set of coordinates are selected to be the initial point of the integrations.
2. The coordinates are passed to the routine **RK78** which returns the subsequent guess.
3. This guess is iteratively fed to the routine **RK78** till the integrations reach the pre-decided final time.

4.3.1 VFSSB

This section shows the results of the tests of the routine **VFSSB** and displays results of integration with respect to the Solar System barycenter. The steps followed are as follows:

1. An initial set of coordinates are chosen for the initial date '0'.
2. The coordinates are transformed into equatorial coordinates if they are adimensional.

Table 4.4 Initial Coordinates in Adimensional Reference Frame

$x = -1.011267857487360811$
$y = 0.00$
$z = 9.069633331713512900 \times 10^{-4}$
$v_x = 0.00$
$v_y = 9.073734203283341515{\times}10^{-3}$
$v_z = 0.00$

3. The coordinates are fed to the RK78 routine.
4. The output of the routine is obtained and stored. Output is also converted into adimensional coordinates to see the orbit in adimensional synodical reference frame.
5. The process is repeated till a prechosen final time.

The tests were made for two different cases. The first test was done with a set of adimensional coordinates in Sun-Earth+Moon system shown in Table 4.4 as the initial coordinates. For this case the integrations were performed for 145.33 days (2.5 in adimensional time). The results are shown in Figure H.1 in equatorial coordinates and Figure H.2 in adimensional coordinates.

The second test was in the Sun-Mars system. The coordinates of Phobos on date '0' were used as the initial point. The integrations were performed for 1.09 days (0.01 in adimensional time) and the results are shown in Figure H.3 and in Figure H.4 in adimensional coordinates. The code used can be found in the file *'/main/examples/TEST-VFSSB/main_vfssb.f'*.

4.3.2 VFIBC

The tests for the routine **VFIBC** follow steps that are exactly similar to those of **VFSSB**, the only difference being that the coordinates are now with respect to a frame of reference which has its origin in a body other than the solar system barycenter. Again, the tests were carried out for two cases. The coordinates used for testing were transformed into the relevant reference frame before integrations. The first test was done with a set of adimensional coordinates in Sun-Earth+Moon system shown in Table 4.4 with a reference frame centred in Uranus. The integrations were carried for 145.33 days. The results are shown in Figure H.5 in equatorial coordinates and in Figure H.6 in adimensional coordinates.

The second case was again the coordinates of Phobos transformed into a reference frame centred in Europa. The integrations were carried on for 5.46 days (0.05 in adimensional time) and the results are shown in Figure H.7 in equatorial coordinates and in Figure H.8 in adimensional coordinates. The code used can be found in the file *'/main/examples/TEST-VFIBC/main_ vfibc.f'*.

4.3.3 Comparison of VFSSB and VFIBC

Table 4.5 Relative Errors between Outputs of VFSSB and VFIBC

Centre of Reference Frame	Relative Error
Sun	$x = -7.23600 \times 10^{-8}$ $y = -2.89100 \times 10^{-8}$ $z = -3.34652 \times 10^{-8}$ $v_x = 4.8960 \times 10^{-8}$ $v_y = -1.78691 \times 10^{-7}$ $v_z = -1.83175 \times 10^{-7}$
Venus	$x = 3.96580 \times 10^{-6}$ $y = 1.62244 \times 10^{-6}$ $z = 1.92020 \times 10^{-6}$ $v_x = -2.39401 \times 10^{-6}$ $v_y = 1.02495 \times 10^{-5}$ $v_z = 1.05517 \times 10^{-5}$
Mars	$x = 3.46641 \times 10^{-7}$ $y = 1.42182 \times 10^{-7}$ $z = 1.68214 \times 10^{-7}$ $v_x = -1.89492 \times 10^{-7}$ $v_y = 9.53508 \times 10^{-7}$ $v_z = 9.85966 \times 10^{-7}$

The aim of this test was to make sure that the integrations being performed in two different reference frames lead to the same result. The steps performed are as follows:

1. An initial point is taken with respect to the solar system barycenter
2. This point is transformed to a reference frame with respect to some other body in the solar system with the help of the routine **TRANSIBC**.
3. The integrations are then performed in both the reference frame till the same final time.
4. The final point of the second reference frame (the one centred in some body) is transformed back to the solar system barycentric reference frame.

5. Finally, the relative errors are computed.

The test was performed for three different reference frames centred in Sun, Venus, and Jupiter. Integrations were performed for 145.33 days and the results are shown in Table 4.5. The initial set of coordinates were the same as given in Table 4.4. As can be seen from the results the relative errors are very small and thus it can be concluded that the routines agree with each other. The code used can be found in the file *'/main/examples/TEST-INTIBC/main_vfibc _vfssb.f'*.

4.4 Tests of Variational Equations

The test of variational equations are based on the concept elaborated in Section 2.2. An initial set of coordinates are chosen as those shown in Table 4.4. Another set of coordinates which differ from the initial condition by a vector Δ is also chosen. The initial matrix M is initialised as an identity matrix. Then the vector field is integrated for both the initial conditions till a pre-decided time. The final coordinates obtained after the integrations must satisfy the Equation (2.3).

4.4.1 VFSSB

The tests were performed for two different initial conditions; one in the Sun-Earth+Moon system and one in Sun-Mars system. The perturbation (in equatorial coordinates) considered was given by the vector,

$$\Delta = (-1 \times 10^{-5}, 2 \times 10^{-5}, 3 \times 10^{-5}, -4 \times 10^{-5}, 5 \times 10^{-5}, -6 \times 10^{-5})^T \qquad \textbf{(4.2)}$$

The results are shown in Table H.1. Integrations were performed for 143.33 days in the Sun-Earth+Moon system and for 2.18 days in Sun-Mars system. The code used can be found in the folder *'/main/examples/TEST-VARIATIONAL/main_variational.f'*.

4.4.2 VFIBC

The tests were performed for two different reference frames; one centred in the Mercury and one centred in Sun. Integrations were performed for 143.33 days in both the cases. The perturbation (in equatorial coordinates) considered was given by the vector,

$$\Delta = (0.1 \times 10^{-4}, 0.1 \times 10^{-6}, 0.2 \times 10^{-3}, 0.3 \times 10^{-4}, 0.4 \times 10^{-2}, 0.1 \times 10^{-2})^T \qquad \textbf{(4.3)}$$

The results are shown in Table H.2. The code used can be found in the folder *'/main/examples/TEST-VARIATIONALIBC/main_variational.f'*.

Chapter 5

APPLICATIONS

5.1 Coefficients of the Lagrangian Form

One of the applications of the package is to compute the coefficients of the Lagrangian and Hamiltonian form of the restricted n-body problem in adimensional synodic reference frame and to implement the quasi periodic models of the solar system. What is important to note is that the coefficients derived in Sections 1.6, 1.7, and 1.8 are functions of time. This dependency on the time can be used to see their variation with time and these coefficients tend to be quasi periodic.

Fourier Analysis of these coefficients was intended to be a part of this thesis but was later abandoned for fixing the package as some problems were encountered with the older version. Nevertheless, routines for computations of the coefficients were designed. These routines have been described in the next section.

5.2 Routines for Lagrangian and Hamiltonian Formulations

This section lists the description of routines that can be used for computation of coefficients of Lagrangian, Hamiltonian, and the quasi periodic formulation of the restricted n-body problem in adimensional synodic reference frame.

SUBROUTINE COEFFLAGRANGIAN(P1,P2)

1. REAL*8 P1: The time in equatorial J2000 reference frame
2. REAL*8 P2(15): An empty array of length '15' (for 15 coefficients).

This routine initialises the value of fifteen different coefficients required for the computation of the value of Lagrangian for restricted n-body problem in adimensional synodic reference frame. The routine uses the equations as specified in the Section 1.6. This routine uses the routine **CALCPARAMRTBP** to calculate various parameters required for the computation of the coefficients. The coefficients are returned in this array. A call to the routine looks like,

CALL COEFFLAGRANGIAN(1000.D0, COEFFA)

If, for example, we need the coefficient a_5 then we can access it at the array index '5' i.e. at *COEFFA(5)*. It is an internal routine and is used by a lot of other routines.

SUBROUTINE LAGRANGIANRTBP(P1,P2,P3,P4)

1. REAL*8 P1: The time in equatorial J2000 reference frame

2. REAL*8 P2(6): Input containing position and velocity (in adimensional coordinates) of the point where Lagrangian has to be computed

3. INTEGER P3: an unused integer input ‘N’

4. REAL*8 P4: Empty variable to hold the value of the Lagrangian

This routine computes the value of Lagrangian for restricted n-body problem in adimensional synodic reference frame. The routine uses the equations as specified in the Section 1.6. This routine calls the routine **COEFFLAGRANGIAN** to calculated various coefficients required for the computation of the coefficients. A call to the routine looks like,

CALL LAGRANGIANRTBP(1000.D0, X, 0, LGN)

In this example, the routine will return the value of the Lagrangian at the position and velocity mentioned in the array X. The output will be stored in the variable *LGN*. As the variable *N* is unused we supplied a random value ‘0’ to it. The reason for having this unused variable is that this preserves the form similar to the routine **VFSSB**.

SUBROUTINE COEFFHAMILTONIAN(P1,P2)

1. REAL*8 P1: The time in equatorial J2000 reference frame

2. REAL*8 P2(15): An empty array of length ‘15’(for 15 coefficients) to hold the output.

This routine initialises the value of fifteen different coefficients required for the computation of the value of Hamiltonian for restricted n-body problem in adimensional synodic reference frame. The routine uses the equations as specified in the Section 1.7. This routine uses the routine **COEFFLAGRANGIAN** to calculate various Lagrangian coefficients required for the computation of the Hamiltonian coefficients.

A call to the routine looks like,

CALL COEFFHAMILTONIAN(1000.D0, COEFFB)

If, for example, we need the coefficient b_5 then we can access it at the array index ‘5’ i.e. at *COEFFB(5)*

SUBROUTINE HAMILTONIANRTBP(P1,P2,P3,P4)

1. REAL*8 P1: The time in equatorial J2000 reference frame

2. REAL*8 P2(6): A vector of length 6 containing position and momentum (in adimensional coordinates) of the point where Hamiltonian has to be computed,

3. INTEGER P3: An unused integer input *N*

4. REAL*8 P4: An empty REAL variable to hold the value of the Hamiltonian

This routine computes the value of Hamiltonian for restricted n-body problem in adimensional synodic reference frame. The routine uses the equations as specified in the Section 1.7. This routine calls the routine **COEFFHAMILTONIAN** to calculated various coefficients required for the computation of the Hamiltonian. A call to the routine looks like,

CALL HAMILTONIANRTBP(1000.D0, X, 0, HMT)

In this example, the routine will return the value of the Hamiltonian at the position and velocity mentioned in the array X. The output will be stored in the variable *HMT*.

SUBROUTINE COEFFQUASI(P1,P2,P3)

1. REAL*8 P1: The time in equatorial J2000 reference frame
2. REAL*8 P2(13): An empty array of length '13' (for 13 coefficients) to hold the output.
3. REAL*8 P3(13): An empty array of length '13' (for 13 derivatives of the coefficients)

This routine initialises the value of thirteen different coefficients required for the computation of the quasi periodic equations discussed in Section 1.8. This routine uses the routine **COEFFLAGRANGIAN** to calculate generate Lagrangian coefficients required for the computation of these quasi coefficients. A call to the routine looks like,

CALL COEFFQUASI(T, COEFFC, COEFFCD)

If, for example, we need the coefficient c_5 then we can access it at the array index '5' i.e. at *COEFFC(5)*. This routine is internal at this point but it might come in handy for the user if Fourier analysis is to be performed.

SUBROUTINE CALCOMEGAQUASI(P1,P2,P3,P4)

1. REAL*8 P1: The time in equatorial J2000 reference frame
2. REAL*8 P2(6): Adimensional coordinates in form of a one dimensional array of length six (3 for position, 3 for velocity)
3. REAL*8 P3: Empty variable to receive the output of the function
4. INTEGER P4: An integer request variable

This routine computes the value of Omega and other values as mentioned in the Section 1.8 and Annexure D. It is required by other routines and therefore is an internal routine. The output of the function depend on the request variable. The following table lists the request variable and the corresponding output.

A call to the routine looks like,

CALL CALCOMEGAQUASI(1000.D0, X, OMEGA, 13)

In this example, the routine will return the value of the $\frac{\partial^2\Omega}{\partial x\partial z}$ in the variable *OMEGA* calculated at the coordinates supplied in the at the position and velocity mentioned in the array X at an adimensional time 1000.

Table 5.1 Values Returned by CALCOMEGAQUASI

Request Variable	Output
0	Ω
1	$\frac{\partial \Omega}{\partial x}$
2	$\frac{\partial \Omega}{\partial y}$
3	$\frac{\partial \Omega}{\partial z}$
4	$\frac{\partial \Omega}{\partial t}$
11	$\frac{\partial^2 \Omega}{\partial x^2}$
12	$\frac{\partial^2 \Omega}{\partial x \partial y}$
13	$\frac{\partial^2 \Omega}{\partial x \partial z}$
14	$\frac{\partial^2 \Omega}{\partial x \partial t}$
21	$\frac{\partial^2 \Omega}{\partial y \partial x}$
22	$\frac{\partial^2 \Omega}{\partial y^2}$
23	$\frac{\partial^2 \Omega}{\partial y \partial z}$
24	$\frac{\partial^2 \Omega}{\partial y \partial t}$
31	$\frac{\partial^2 \Omega}{\partial z \partial x}$
32	$\frac{\partial^2 \Omega}{\partial z \partial y}$
33	$\frac{\partial^2 \Omega}{\partial z^2}$
34	$\frac{\partial^2 \Omega}{\partial z \partial t}$

SUBROUTINE QUASIHAMILTONIAN(P1,P2,P3,P4)

1. REAL*8 P1: The time in equatorial J2000 reference frame
2. REAL*8 P2(48): Adimensional coordinates in form of a one dimensional array of length 48 (3 for position, 3 for velocity, rest for variational equations
3. INTEGER P3: Integer variable *N* specifying whether only the vector field is needed or the variational equations are needed as well. The variable *N* can have two values: '6' means only the vector field are returned, '48' means the vector field as well as the variational equations are computed and returned.
4. REAL*8 P4: An empty array of length 48 to hold the output of the function

This routine computes the value of quasi-periodic vector field at a given point. An example of a call to the routine looks like,

CALL QUASIHAMILTONIAN(1000.D0, X, 48, F)

In this example , the routine will compute the value of the variational equations and the vector fields on the date '1000' based on the positions supplied in array *X* and will return the output in the array *F*.

5.3 Tests of QUASIHAMILTONIAN

The routine **QUASIHAMILTONIAN** gives a quasi-periodic vector field. This vector field can be integrated in and the routine can also produce variational equations. Both these aspects of the routines were tested and implemented. The results and procedure are shown in the following subsection.

5.3.1 Integration

The integrations in this routine were performed with the initial coordinates of Phobos on the date '0' in the Sun-Mars system. The integrations were performed for 2.18×10^{-02} days. The results of integrations are shown in the Figure H.9 in equatorial coordinates and in Figure H.10 in adimensional coordinates. The code used can be found in the file *'/main/examples/TEST-INTQUASI/main_int_quasi.f'*.

5.3.2 Variational Equations

The concept of the test of variational equations using **QUASIHAMILTONIAN** is the same as elaborated in Section 4.4. The tests were performed for two different initial conditions; one in the Sun-Earth+Moon system and one in Sun-Mars system. For the Sun-Earth+Moon system the initial coordinates were the same as given by Table 4.4 and for the Sun-Mars system the coordinates for Phobos were obtained for the date '0' in J2000 equatorial time. The integrations for the case of Sun-Earth+Moon system was carried on for 5.23 days and the integrations for Sun-Mars system was carried on for 9.84 days. The perturbation (in adimensional coordinates) considered was given by the vector,

$$\Delta = (-1 \times 10^{-5}, 2 \times 10^{-5}, 3 \times 10^{-5}, -4 \times 10^{-5}, 5 \times 10^{-5}, -6 \times 10^{-5})^T \tag{5.1}$$

The results are shown in Table 5.2. The code used can be found in the folder *'/main/examples/TEST-VARIATIONALQUASI/main_variational_quasi.f'*.

Table 5.2 Maximum Relative Errors for Variational Equations from QUASIHAMILTONIAN

Initial Coordinates	Relative Error
$x = -1.0112678574873608111$	$x = 3.24701 \times 10^{-6}$
$y = 0.00$	$y = 1.69323 \times 10^{-6}$
$z = 9.0696333317135129000 \times 10^{-4}$	$z = 1.92545 \times 10^{-5}$
$v_x = 0.00$	$v_x = 2.16546 \times 10^{-6}$
$v_y = 9.0737342032833415 15 \times 10^{-3}$	$v_y = 8.15063 \times 10^{-5}$
$v_z = 0.00$	$v_z = 6.93009 \times 10^{-6}$
$x = -1.0000239850977075$	$x = -5.61078 \times 10^{-3}$
$y = -3.5625419288151113 2 \times 10^{-3}$	$y = -1.60334 \times 10^{-8}$
$z = -1.0538753708409903 0 \times 10^{-2}$	$z = -2.88880 \times 10^{-2}$
$v_x = 7.1695640348528988 4 \times 10^{-2}$	$v_x = 7.60705 \times 10^{-6}$
$v_y = -5.8413423714112976 1 \times 10^{-2}$	$v_y = -6.09580 \times 10^{-8}$
$v_z = 3.3693947888367444 0 \times 10^{-2}$	$v_z = 3.73097 \times 10^{-2}$

5.4 Trajectory Refinement

One of the application of this package that we consider for this thesis is the refinement of libration point orbits in two systems: Jupiter-Europa and Saturn-Titan. This section will elaborate the generation of the initial guess for orbit refinement, the concept of parallel shooting, and will finally present the results in the two aforementioned systems.

5.4.1 Parallel Shooting

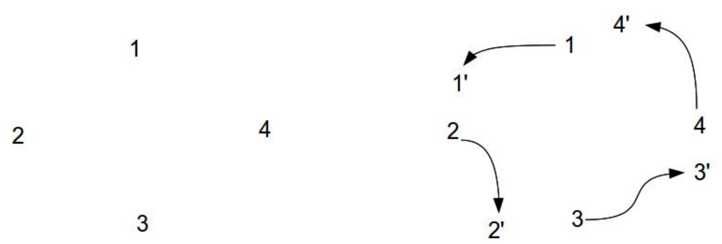


Figure 5.1 Concept of Parallel Shooting

This section describes trajectory refinement using the concept of parallel shooting. For numerical computation of orbit in a time period *T* we first divide this period into *N* parts with steps of Δt. Suppose the state at any point *i*, of these *N* points, Q_i can be stated as,

$$Q_i = (t_i, x_i, y_i, z_i, \dot{x}_i, \dot{y}_i, \dot{z}_i)^T \tag{5.2}$$

Now, we define the term $\phi(Q_i)$ as the image of the point Q_i integrated in a vector field for the time Δt. It is easy to see that the point $\phi(Q_i)$ must be same as the point Q_{i+1}. As an example, we assume that our periodic orbit has been divided into four parts as shown as in Figure 5.1.

When we integrate these points points for time Δt then the image of point **1** is **1'**, the image of point **2** is **2'** and so on and so forth. In this example, the point **1'** must be the same as the point **2'**. Now suppose that if $\phi(Q_i) = Q_{i+1}$ for $i = 1, 2, \ldots, N-1$ then we obtain our final refined orbit else we need to change the starting positions to fulfil this condition. This will give us $N-1$ non-linear equations, which can be expressed as [10] [15],

$$F\begin{pmatrix} Q_1 \\ Q_2 \\ \vdots \\ Q_N \end{pmatrix} = \begin{pmatrix} \phi(Q_1) \\ \phi(Q_2) \\ \vdots \\ \phi(Q_N - 1) \end{pmatrix} - \begin{pmatrix} Q_2 \\ Q_3 \\ \vdots \\ Q_N \end{pmatrix} = \phi\begin{pmatrix} Q_1 \\ Q_2 \\ \vdots \\ Q_N - 1 \end{pmatrix} - \begin{pmatrix} Q_2 \\ Q_3 \\ \vdots \\ Q_N \end{pmatrix} = 0 \tag{5.3}$$

Newton's method is used to solve this set of equations. If $Q^{(j)} = (Q_1^{(j)}, Q_2^{(j)}, \ldots, Q_N)^{(j)})^T$ represents the *j*th iteration of the Newton's method then we can express the Newton's equation as [10],

$$DF(Q^{(j)})\,(Q^{(j+1)} - Q^{(j)}) = -F(Q^{(j)}) \tag{5.4}$$

where the matrix *DF* is given by,

$$DF = \begin{pmatrix} A_1 & -I & & & \\ & A_1 & -I & & \\ & & \ddots & \ddots & \\ & & & A_{N-1} & -I \end{pmatrix} \tag{5.5}$$

and,

$$D\phi = \begin{pmatrix} A_1 & & & \\ & A_1 & & \\ & & \ddots & \\ & & & A_{N-1} \end{pmatrix} \tag{5.6}$$

The transition matrices, A_i, are 6 by 6 at each step of the method. Thus, we get $(N-1)\times 6$ number of equations and the number of unknowns is N. Since we have more unknowns that the number of equations, we will not get a solution but a hyperplane of solutions. Now, from these solutions the solution must be chosen such that the final orbit is closer to the initial (seed) orbit. The 'closeness' will be judged by using a prefixed condition: evaluation of the norm of the error.

5.4.2 Initial Guess for Parallel Shooting

The initial guess for parallel shooting routines is obtained by first expanding the equations of motion and then using the Lidnstedt-Poincaré method to solve these equations. For the purpose of refinement of libration point orbits we use the restricted three body model from Equations (1.74), (1.75), and (1.76). We can transform coordinates to libration point coordinate system described in [13]. Thus, the equations of motion of the restricted three body problem can be expressed as [14] [13],

$$\ddot{x} - 2\,\dot{y} = \frac{1}{\gamma^2}\Omega_x \tag{5.7}$$

$$\ddot{y} + 2\,\dot{x} = \frac{1}{\gamma^2}\Omega_y \tag{5.8}$$

$$\ddot{z} = \frac{1}{\gamma^2}\Omega_z \tag{5.9}$$

To expand the values of Ω_x, Ω_y, and Ω_z we use Legendre Polynomials which are given as,

$$\begin{aligned} \frac{1}{\sqrt{(x-A)^2 + (y-B)^2 + (z-C)^2}} &= \frac{1}{\sqrt{A^2 + B^2 + C^2}} \\ \sum_{n=0}^{\infty} \left(\frac{\rho}{\sqrt{A^2 + B^2 + C^2}} \right) &. P_n \left(\frac{Ax + By + Cz}{\sqrt{A^2 + B^2 + C^2}} \right) \end{aligned} \tag{5.10}$$

where,

$$\rho^2 = x^2 + y^2 + z^2 \tag{5.11}$$

By further calculation, which are out of scope of this thesis, we obtain [14],

$$\ddot{x} - 2\,\dot{y} - (1 + 2\,c_2)\,x = \frac{\partial}{\partial x} \sum_{n \geq 3} c_n \rho^n P_n \left(\frac{x}{\rho} \right) \tag{5.12}$$

$$\ddot{y} + 2\,\dot{x} + (c_2 - 1)\,y = \frac{\partial}{\partial y} \sum_{n \geq 3} c_n \rho^n P_n \left(\frac{x}{\rho} \right) \tag{5.13}$$

$$\ddot{z} + c_2\,z = \frac{\partial}{\partial x} \sum_{n \geq 3} c_n \rho^n P_n \left(\frac{x}{\rho} \right) \tag{5.14}$$

Here, ρ is given by Equation (5.11) and the coefficients are expressed as [13],

$$c_n = \frac{1}{\gamma^3} \left((\pm 1)^n \mu + (-1)^n \frac{(1-\mu)\,\gamma_j^{n+1}}{(1 \mp \gamma_j)^{n+1}} \right), \text{ for } L_j\text{, j=1,2} \tag{5.15}$$

and,

$$c_n = \frac{(-1)^n}{\gamma_3^3} \left(1 - \mu + \frac{\mu\,\gamma_3^{n+1}}{(1+\gamma_3)^{n+1}} \right), \text{ for } L_3 \tag{5.16}$$

where, L_1, L_2, and L_3 are three libration points of the system.

5.4.3 Lissajous Orbit

As stated before, now that we have semi-analytical equations we can use the Lindstedt-Poincaré method to solve them to obtain a Lissajous orbit around a libration point. First, we need a starting point for the computation, we use the linear part of Equations (5.12), (5.13), and (5.14) as,

$$\ddot{x} - 2\,\dot{y} - (1 + 2\,c_2)\,x = 0 \tag{5.17}$$

$$\ddot{y} + 2\,\dot{x} + (c_2 - 1)\,y = 0 \tag{5.18}$$

$$\ddot{z} + c_2\,z = 0 \tag{5.19}$$

This can be expressed as [13],

$$x(t) = \alpha\,cos\,(\omega_0\,t + \phi_1) \tag{5.20}$$

$$y(t) = \kappa\alpha\,cos\,(\omega_0\,t + \phi_1) \tag{5.21}$$

$$z(t) = \beta\,cos\,(\nu_0\,t + \phi_2) \tag{5.22}$$

where,

$$\omega_0 = \sqrt{\frac{2 - c_2 + \sqrt{9\,c_2^2 - 8\,c_2}}{2}} \tag{5.23}$$

$$\nu_0 = \sqrt{c_2} \tag{5.24}$$

$$\kappa = \frac{-\,(\omega_0 + 1 + 2\,c_2)}{2\,\omega_0} \tag{5.25}$$

The parameterα is the in plane amplitude, β is the out of plane amplitude, and ϕ_1 and ϕ_2 are the phases. Now when we add the non linear terms in the Equations (5.12), (5.13), and (5.14) we have to look for solutions of the form,

$$x(t) = \sum_{i,j=1}^{\infty}\left(\sum_{|k|\le i,|m|\le j} x_{ijkm}\,cos\,(k\,\theta_1 + m\,\theta_2)\right)\alpha^i\,\beta^j \tag{5.26}$$

$$y(t) = \sum_{i,j=1}^{\infty}\left(\sum_{|k|\le i,|m|\le j} y_{ijkm}\,sin\,(k\,\theta_1 + m\,\theta_2)\right)\alpha^i\,\beta^j \tag{5.27}$$

$$z(t) = \sum_{i,j=1}^{\infty}\left(\sum_{|k|\le i,|m|\le j} z_{ijkm}\,cos\,(k\,\theta_1 + m\,\theta_2)\right)\alpha^i\,\beta^j \tag{5.28}$$

where,

$$\theta_1 = \omega\,t + \phi_1, \quad \theta_2 = \nu\,t + \phi_2 \tag{5.29}$$

$$\omega = \sum_{i,j=0}^{\infty} \omega_{ij}\,\alpha^i\,\beta^j, \quad \nu = \sum_{i,j=0}^{\infty} \nu_{ij}\,\alpha^i\,\beta^j \tag{5.30}$$

The Lindstedt-Poincaré methodology provides the computations of the coefficients x_{ijkm}, y_{ijkm}, z_{ijkm}, ω_{ij}, and ν_{ij}, with a perturbative approach. These coefficients were obtained using program designed as part of the work in Reference [13]. They were stored in a file and was used in inputs for routine **DIBLIS** to generate Lissajous orbits.

5.4.4 Halo Orbits

The equations of motion specified in Equations (5.12), (5.13), and (5.14) can be re-written as [13],

$$\ddot{x} - 2\,\dot{y} - (1 + 2\,c_2)\,x = \sum_{n\geq 2} c_{n+1}\,(n+1)\,T_n \tag{5.31}$$

$$\ddot{y} + 2\,\dot{x} + (c_2 - 1)\,y = y \sum_{n\geq 2} c_{n+1}\,R_{n-1} \tag{5.32}$$

$$\ddot{z} + c_2\,z = z \sum_{n\geq 2} c_{n+1}\,R_{n-1} \tag{5.33}$$

where,

$$T_n\,(x, y, z) = \rho^n\,P_n\left(\frac{x}{\rho}\right) \tag{5.34}$$

$$R_{n-1}\,(x, y, z) = \frac{1}{y}\,\frac{\partial T_{n+1}}{\partial y} \tag{5.35}$$

also,

$$R_{n-1}\,(x, y, z) = \frac{1}{1}\,\frac{\partial T_{n+1}}{\partial z} \tag{5.36}$$

$$\frac{\partial T_{n+1}}{\partial z} = (n+1)\,T_n \tag{5.37}$$

These equations have to be changed to express the equations of halo orbits. We add a term Δz to the Equation (5.33), where,

$$\Delta = \sum_{i,j=0}^{\infty} d_{ij}\,\alpha^i\,\beta^j \tag{5.38}$$

The coefficient d_{ij} has to be computed numerically and iteratively. Thus, Equation (5.33) can be re-written as,

$$\ddot{z} + c_2\,z = z \sum_{n\geq 2} c_{n+1}\,R_{n-1} + \Delta\,z \tag{5.39}$$

The term Δz lets us force the out-of-plane frequency to be the same as the in-plane one (as it must be for a periodic orbit). But then to keep the equation in the initial form $\Delta = 0$ is required, so, in fact, Equation (5.38) represents a constraint between the amplitudes α and β. The starting solution for the Lindstedt-Poincaré for the Halo orbit can be used as Equations (5.17), (5.18), and,

$$\ddot{z} + c_2\,z = d_{00}\,z \tag{5.40}$$

where,

$$d_{00} = c_2 - \omega_0^2 \tag{5.41}$$

The solution of the equations can be expressed as,

$$x(t) = \alpha\, cos\,(\omega_0\, t + \phi) \tag{5.42}$$
$$y(t) = \kappa\, \alpha\, cos(\omega_0\, t + phi) \tag{5.43}$$
$$z(t) = \beta\, cos\,(\omega_0\, t + \phi) \tag{5.44}$$

The parameter κ is the same as given by Equation (5.25), ϕ is the phase, and ω_0 is given as,

$$\omega_0 = \sqrt{2 - c_2 + \frac{\sqrt{9\, c_2^2 - 8\, c_2}}{2}} \tag{5.45}$$

We look for the expansions of the form,

$$x(t) = \sum_{i,j=1}^{\infty} \left(\sum_{|k| \le i+j} x_{(i\,j\,k)}\, cos\,(k\,\theta) \right) \alpha^i\, \beta^j \tag{5.46}$$

$$y(t) = \sum_{i,j=1}^{\infty} \left(\sum_{|k| \le i+j} y_{(i\,j\,k)}\, sin\,(k\,\theta) \right) \alpha^i\, \beta^j \tag{5.47}$$

$$z(t) = \sum_{i,j=1}^{\infty} \left(\sum_{|k| \le i+j} z_{(i\,j\,k)}\, cos\,(k\,\theta) \right) \alpha^i\, \beta^j \tag{5.48}$$

As with the case of Lissajous orbits, the Lindstedt-Poincaré methodology provides the computations of the coefficients x_{ijk}, y_{ijk}, z_{ijk}, ω_{ij}, and d_{ij}. These coefficients were again obtained using program designed as part of the work in Reference [13] and were stored in a file. This file was used as input for routine **DIBHAL** to generate Halo orbits.

5.4.5 Orbit Generators

DIBHAL

This routine that can be used to generate halo Orbit in Jupiter-Europa orbit around L2. It is located in the folder '$/parallelshooting/orbit - generators/halo/$'. The user can produce the sample orbit by first going to the directory and compiling the routine using the '$./make.csh$' script and then executing the executable file produced. The output is written in a file named '*rtbp-orbit.dat*' and is stored in the same directory. It is important to note that the orbit is in adimensional RTBP reference frame with respect to the solar system barycenter.

The output of this routine can be copied to the folder '$/parallelshooting/$' and can be used as input to the routine **CHANGECOORINDATES**.

DIBLIS

The routine that can be used to generate Lissajous Orbit in Saturn-Titan System about L2. It is located in the folder '$/parallelshooting/orbit-generators/liss/$'. The user can produce the orbit by first entering this directory and compiling the routine using the '$./make.csh$' script and then executing the executable file produced. The output is stored in a file named '*rtbp-orbit.dat*' and is stored in the same directory. Again, it is important to note that the orbit is in adimensional RTBP reference frame with respect to the solar system barycenter. The output of this routine can be copied to the folder '$/parallelshooting/$' and can be used as input to the routine **CHANGECOORINDATES**.

5.4.6 Parallel Shooting Routines

The parallel shooting package has been provided with the program designed as a part of this thesis. The directory structure of the program has been shown in Figure 3.1. The directory '$/parallelshooting/orbit-generators/$' contains code to generate two different orbits; one halo and one Lissajous. The root directory '$/parallelshooting/$' contains important routines which have been described in the following subsections.

TIRP and TIRP_IBC

This routine is the primary routine that can be used to perform parallel shooting and refinement of orbits. **TIRP** performs integrations with respect to the solar system barycenter whereas **TIRP_IBC** performs integrations with respect to any body in the solar system. It requires the output file produced by the routine **CHANGECOORDINATES** as an input. The seed orbit can be in any type of coordinates such as the equatorial, adimensional, or the coordinates in Lagrangian reference system defined in Reference [13]. This routine will ask user for the input file and other parameters like the coordinate reference frame of the output and the time step in terms of days required for the integrations. The output is stored in the file name '$/parallelshooting/dibx.dat$' in terms of seven columns: one column for time, six columns for the components of the positions (three positions and three velocities). It is important to note that if the user wants to perform parallel shooting using **TIRP_IBC** then the file '$/parallelshooting/tirp\_ibc.f$' must be edited to point to initialise the particular *IBC* value. Particularly, the user must look for line label *999* and change it. The line is a call to the routine **NMODJPL**.

CHANGECOORDINATES

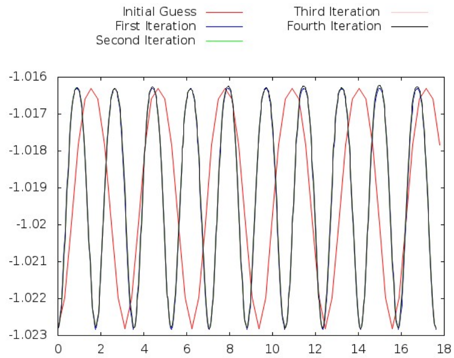
This routine takes an input of a seed orbit in the RTBP adimensional reference frame and generates the input for the parallel shooting routine **TIRP** and **TIRP_IBC**. The input file must have seven columns each containing the time, and six components of the state (three positions, three velocities) for the nodes of the seed orbit . The routine asks the user for the details of the input for example, the name of the file, the RTBP system that the coordinates are referred to and the origin of the original equatorial reference frame. It also provides an option to the user to transform the reference frame to equatorial reference frame or to change to a reference frame situated in some other body in the solar system. After taking these inputs from the user it produces the output file named '*./output_change.dat*' which has to be used as an input to the routine **TIRP** or **TIRP_IBC** depending on the reference frame.

To use the routine the user must execute the script called '*./make_change.csh*' and this will generate an executable '*./make-change.exe*'. It is important to note that the standard length of the input file for this routine has been specified as '101' in this routine. Thus, if the number of nodes for the input file is different from this, the user must open the file '*./changecoordinates.f*' and change the value of the parameter *FILELENGTH* to the length of the input file. The user must then re-compile the routine.

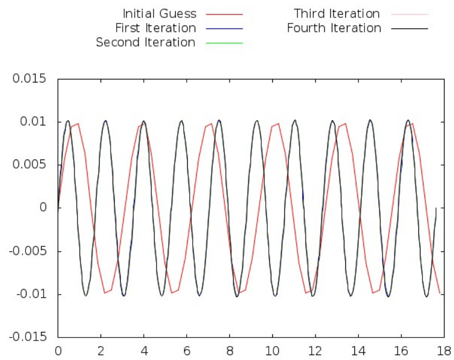
5.5 Parallel Shooting Examples

Figure 5.2 Variation of Axis for Iterations of Parallel Shooting in Jupiter-Europa System

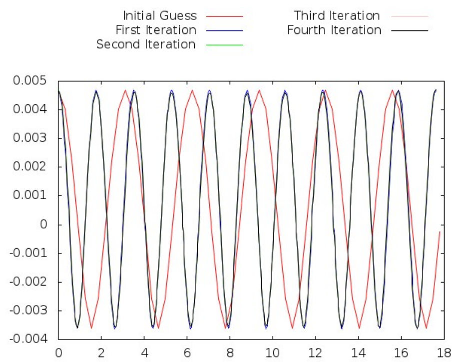
(a) X axis



(b) Y axis



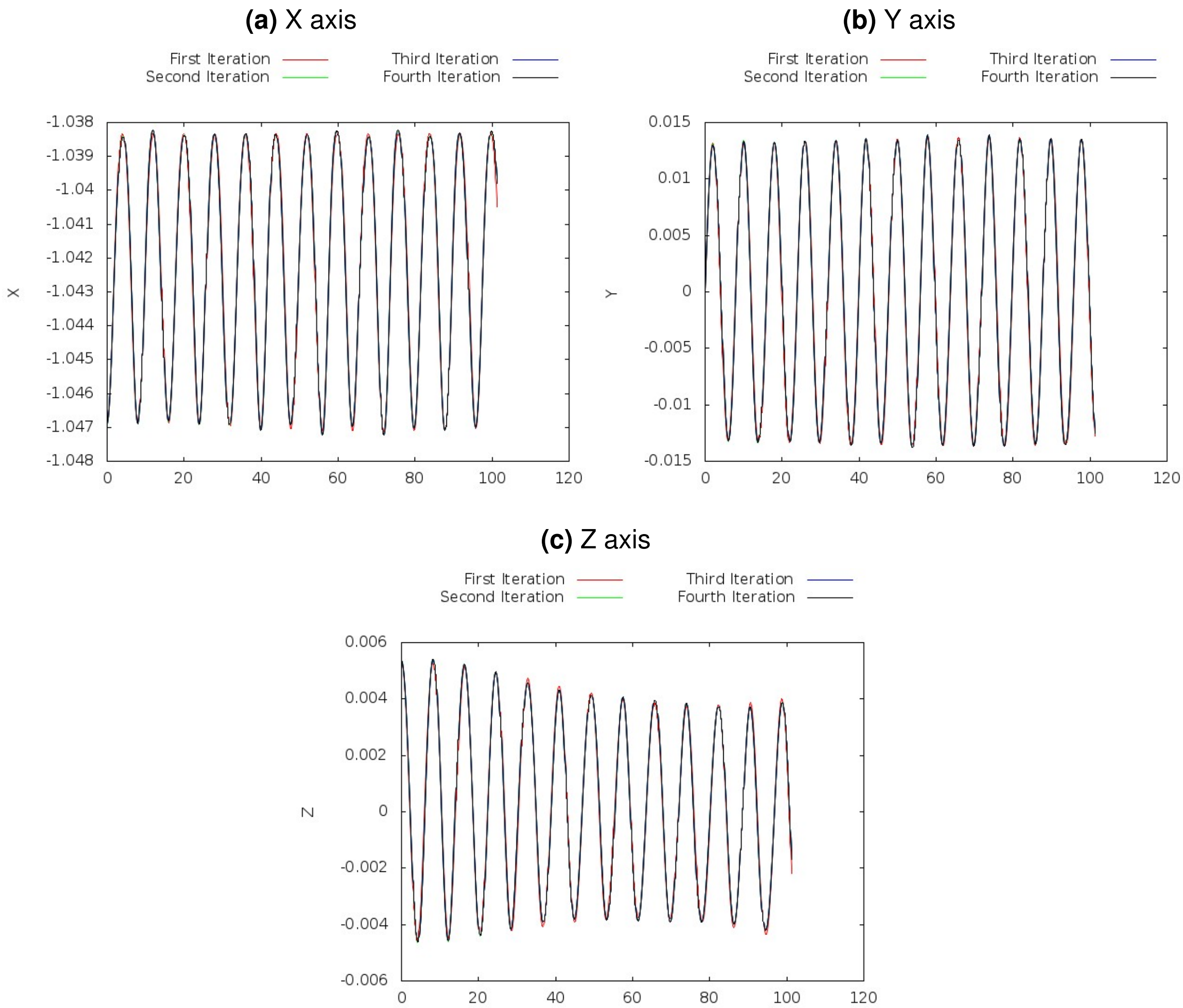
(c) Z axis



5.5.1 Halo Orbit in Jupiter Europa System about L2

The first example was using parallel shooting to refine a halo orbit around the L2 (Lagrangian) point in the Jupiter-Europa system. The calculations were performed with respect to the solar system barycenter in the restricted three body adimensional reference frame. Thus the routine **TIRP** was used for parallel shooting which in turn uses the routine **VFSSB**. The iterations of the parallel shooting is shown in Figure 5.4. Also, the variation of X, Y, and Z axis with adimensional time for different iterations is shown in Figure 5.2. In both the figures the iterations are mostly overlapping.

Figure 5.3 Variation of Axis for Iterations of Parallel Shooting in Saturn-Titan System

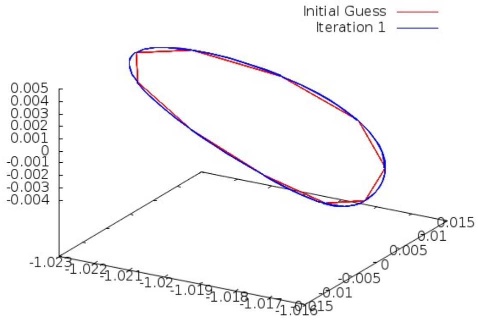


5.5.2 Lissajous Orbit in Saturn-Titan System about L2

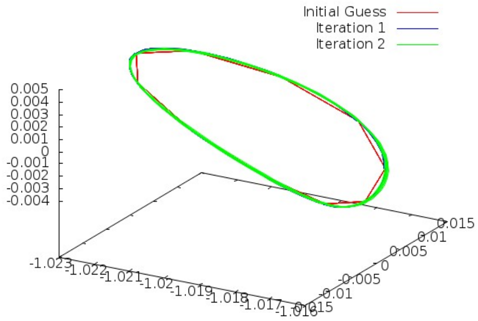
The second example was using parallel shooting to refine a Lissajous orbit around the L2 (Lagrangian) point in the Saturn-Titan system. The calculations were performed with respect to a reference frame situated in Jupiter. Thus, the routine **TIRP_IBC** was used for parallel shooting which in turn uses the routine **VFIBC**. Since we were working with adimensional coordinates, the use of a reference frame will not produce a difference in the output but since Jupiter is closer to the Saturn-Titan system, it can reduce the computations in terms of transformations of coordinates as **VFIBC** takes equatorial coordinates as an input. The iterations of the parallel shooting is shown in Figure 5.5. Also, the variation of X, Y, and Z axis with adimensional time for different iterations is shown in Figure 5.3. Again, in both the figures the iterations are mostly overlapping.

Figure 5.4 Successive Iterations of Parallel Shooting in Jupiter-Europa System

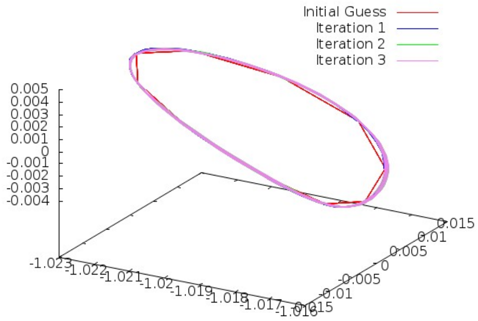
(a) Initial Guess and First Iteration



(b) Second Iteration



(c) Third Iteration



(d) Fourth Iteration

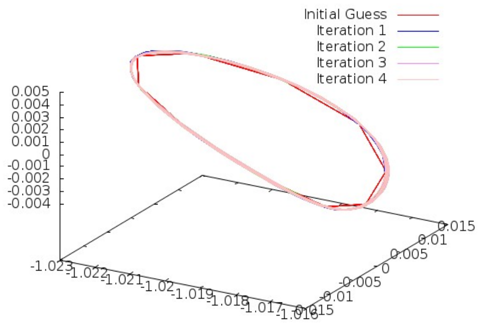
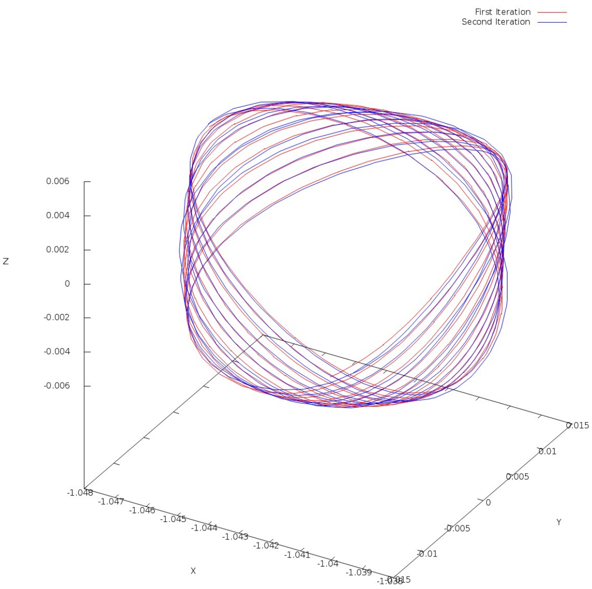
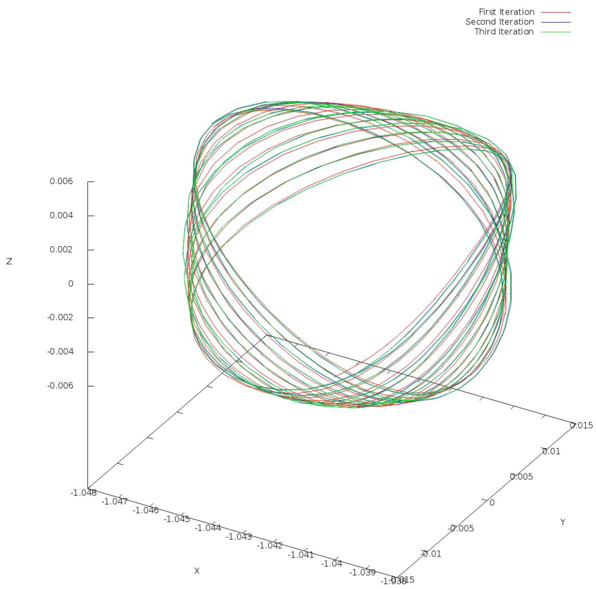


Figure 5.5 Successive Iterations of Parallel Shooting in Saturn-Titan System

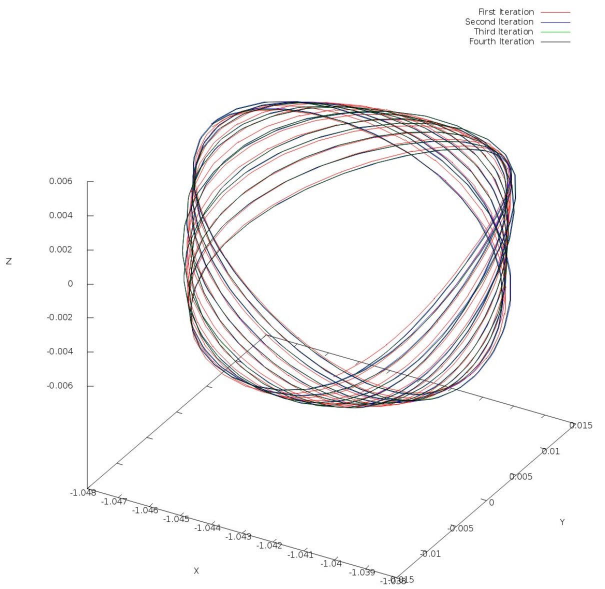
(a) First Iteration and Second Iteration



(b) Third Iteration



(c) Fourth Iteration



CONCLUSION

6.1 Present Work

This chapter presents the principal achievements of this master's thesis. During the thesis a stable package was developed which has the ability to:

1. read numerical ephemeris by accessing SpiceLib from JPL.

2. transform coordinates between equatorial to adimensional synodic system and between different equatorial reference frames in the solar system.

3. perform numerical integrations to obtain orbits in the solar system w.r.t. the solar system barycenter and any other inertial body centered frame. Also, it has ability to perform numerical integrations in the quasi-periodic semi-analytical RTBP like formulation of the restricted n-body problem.

4. return the values of State Transition Matrix using the variational equations. This can be used to observe the change in orbits due to perturbations in three domains: orbits w.r.t. solar system barycenter, w.r.t. any inertial body centered frame, and in the quasi-periodic formulation of the restricted n-body problem.

5. perform trajectory refinement of Halo and Lissajous orbits using parallel shooting. If the user can generate seed orbits for other sytems, then this program can perform parallel shooting in those domains as well.

6. compute the coefficients and values of the Lagranian and Hamiltonian formulation of restricted n-body problem

7. compute the coefficients of the quasi-periodic semi-analytical RTBP like formulation of the restricted n-body problem

6.2 Future Work

Some of the steps that can be performed as future work following this thesis:

1. Fourier analysis of the coefficients of the the quasi-periodic semi-analytical RTBP like formulation of the restricted n-body problem and the Lagrangian and Hamiltonian formulations of the same problem can be performed. This can be used to generate numerous intermediate models between the analytical model and the simplified RTBP model.

2. Analytical values of the over-over and, if possible, over-over-over accelerations can be obtained. Currently, it is calculated numerically and this part can be improved by going into the numerical ephemeris and analytically obtaining the expressions.

3. Complete implementation of the gravity model can be performed. The idea has been presented in this thesis but it has not been implemented yet. Once implemented it can provide immense flexibility to the user to control the gravitational model in real-time.

4. The user friendliness of the package can be improved.

BIBLIOGRAPHY

1. R. R. Bate, D. D. Mueller, and J. E. White, ”Orbit Determination from Observations”, Chapter 2 in *Fundamentals of Astrodynamics*, pp 51-55, Dover Publications Inc., New York, 1971

2. B. D. Tapley, B. E. Schutz, and G. H. Born, ”The Orbit Problem”, Chapter 2 in *Statistical Orbit Determination*, pp 40-81, Elsevier Academic Press, 2004

3. G. Gomez, A. Jorba, C. Simo, J. J. Masdemont, ”Reference Systems and Equations of Motion”, Appendix B in *Dynamics and Mission Design Near Libration Points Vol. III: Advanced Methods for Collinear Points*, pp 137-143, World Scientific Monograph Series Mathematics - Vol. 4, Singapore, 2001

4. G. Gomez, J. Llibre , R. Martinez , C. Simo, ”Halo Orbits. Analytic and Numerical Study”, Chapter 2 in *Dynamics and Mission Design Near Libration Points Vol. I: Fundamentals: The Case of Collinear Libration Points*, pp 59-140, World Scientific Monograph Series Mathematics - Vol. 2

5. V. Szebehely, ”Description of the Three Body Problem”, Chapter 2 in *Theory of Orbits: The Restricted Problem of Three Bodies*, pp 07-39, Academic Press, 1967

6. J. J. Masdemont, ”Transferència a òrbites halo” in *Estudi i Utilizació de Varientats Invariants en Problemes de Mecànica Celeste*, pp 102-125, PhD Thesis, Departament de Matemàtica Aplicada I, UPC, Barcelona, Spain, 1991, ISBN: 8468982415
DOI: $http://www.tdx.cat/handle/10803/6713$

7. P. S. Martín, ”Introduction”, Chapter 1 in *Obtaining Gravitational Models from Planetary Ephemerides*, , pp 13-22, Master's Degree Thesis for MSc. in Mathematics Engineering, UPC, Barcelona, Spain.
DOI: $http://upcommons.upc.edu/handle/2099.1/15246$

8. ”NAIF SPICE Toolkit Hypertext Documentation”, JPL NASA Website.
URL: $http://naif.jpl.nasa.gov/pub/naif/toolkit_docs/FORTRAN/index.html$

9. P. K. Seidelmann, *The Explanatory Supplement to the Astronomical Almanac* , University Science Books, 2008

10. G. Gomez, J. J. Masdemont, and C. Simo, ”Quasihalo Orbits Associated With Libration Points”, *The Journal of the Astronautical Sciences*, Volume 46, pp 29-34, 1998

11. J. A. Shapiro, ”Classical Mechanics”, pp 16-55, November 17, 2010
URL : $http://www.physics.rutgers.edu/\ shapiro/507/book.pdf$

12. E. Fehlberg, *Classical Seventh-, Sixth-, and Fifth-Order Runge Kutta-Nystrom Formula with Stepsize Control for General Second-Order Differential Equations*, NASA Technical Report TR R-432, Washington D.C., October, 1974

13. A. Jorba, J. J. Masdemont, ”Dynamics in the Center Manifold of the Collinear Points of the Restricted Three Body Problem”, *Physica D*, 132, pp 189-213, 1999

14. E. Herrera-Sucarrat, ”Planar LEO-GEO transfer considering Lyapunov orbits of th e Sun - Earth system”, Chapter 2 in *Study of LEO to GEO transfers via the L1 Sun-Earth or Earth-Moon libration points*, Master’s Thesis, Universitat PolitÃ¨cnica de Catalunya, pp 42-45
DOI: $http://hdl.handle.net/2099.1/6546$

15. J. Stoer, R.Bulirsch, ”Ordinary Differential Equations”, Chapter 7 in *Introduction of Numerical Analysis*, pp 499-535, Springer-Verlag, 1972

Appendix A

LAGRANGIAN AND HAMILTONIAN MECHANICS

A.1 Introduction

The aim of this chapter is to provide a basic introduction to Lagrangian and Hamiltonian Mechanics. Classical Mechanics is very important for applications into trajectory prediction. This chapter will elaborate the concept of Lagrangian and the Hamiltonian and will derive the equations of the same.

A.2 Generalised Coordinates

This section will present the concept of generalised coordinates. Use of generalised coordinates make the expression of Lagrangian easy to formulate. Suppose we have a set of 3n Cartesian coordinates in a three dimensional space. Let us assume that we have a set of 3n generalised coordinates such that all the Cartesian coordinates can be expressed in terms of these generalised coordinates and time. Thus any coordinate in the Cartesian system can be expressed as a function of time and these generalised coordinates as,

$$\vec{r_i} = \vec{r_i}(q_1, q_2, q_3 \ldots, q_{3n}, t) \tag{A.1}$$

Vice-versa, it goes that all the generalised coordinates will be a function of the Cartesian coordinates. Therefore,

$$\vec{q_j} = q_j(\vec{r_1}, \vec{r_2}, \vec{r_3} \ldots, \vec{r_{3n}}, t) \tag{A.2}$$

Now, we can focus what happens if there is a change in one of the Cartesian coordinates. Suppose the generalised coordinates are changed by a small amount δq_j then the change in the Cartesian coordinates (represented here forth as x_k) will be,

$$\delta x_k = \sum_j \frac{\partial x_k}{\partial q_j} \delta q_j \tag{A.3}$$

Similarly, for any change in the Cartesian coordinates, the change in the generalised coordinates is given by,

$$\delta q_j = \sum_k \frac{\partial q_j}{\partial x_k} \delta x_k \tag{A.4}$$

It is important to point out that the change produced due to any change in time coordinate has been considered to be zero.

Now the force can be expressed as a change in potential as,

$$F_k = -\frac{\partial U(x)}{\partial x_k} = -\sum_j \frac{U(x_k(q_j))}{\partial q_j}\frac{\partial q_j}{x_k} = -\sum_j \frac{\partial q_j}{\partial x_k} Q_j \tag{A.5}$$

where Q_j is called Generalised Force given by [11],

$$Q_j = \sum_k F_k \frac{\partial x_k}{\partial q_j} \tag{A.6}$$

A.3 The Lagrangian

We define the Lagrangian *L* of a system as follows,

$$L = T - U \tag{A.7}$$

where, *T* is the kinetic energy of the system and *U* is the potential energy is the system. Let's consider a system given in Cartesian coordinates as,

$$m\,\ddot{x}_i = F_i \tag{A.8}$$

The kinetic energy of this system will be,

$$T = \frac{1}{2} m\,\dot{x}^2 \tag{A.9}$$

and the potential energy is given by,

$$U = -F\,x \tag{A.10}$$

The momentum p_i of a system can be given as,

$$p_i = \frac{\partial T}{\partial \dot{x}_i} \tag{A.11}$$

and, the Force *F* can be calculated by differentiating Equation (A.10) with respect to x_i

$$F = -\frac{\partial U}{\partial x_i} \tag{A.12}$$

Now, we will try to use Lagrangian equations to elaborate their usefulness. Differentiating Equation (A.7) with respect to $\dot{x}_i$ and then with respect to t we get,

$$\frac{d\,\partial L}{dt\,\partial \dot{x}_i} = \frac{d\,\partial T}{dt\,\partial \dot{x}_i} - \frac{d\,\partial U}{dt\,\partial \dot{x}_i} \tag{A.13}$$

Putting the value of p_i from Equation (A.11) and since force is the rate of change of momentum with time we obtain,

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{x}_i} = F_i - \frac{d}{dt}\frac{\partial U}{\partial \dot{x}_i} \quad \textbf{(A.14)}$$

Now differentiating Equation (A.7) with respect to x_i we get,

$$\frac{\partial L}{\partial x_i} = \frac{\partial T}{\partial x_i} - \frac{\partial U}{\partial x_i} \quad \textbf{(A.15)}$$

In this equation if we use the value from Equation (A.12) we get,

$$\frac{\partial L}{\partial x_i} = \frac{\partial T}{\partial x_i} + F_i \quad \textbf{(A.16)}$$

Subtracting Equation (A.14) from Equation (A.16) we obtain,

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{x}_i} - \frac{\partial L}{\partial x_i} = -\frac{d}{dt}\frac{\partial U}{\partial \dot{x}_i} - \frac{\partial T}{\partial x_i} \quad \textbf{(A.17)}$$

As can be seen from Equation (A.10), *U* is not a function of $\dot{x}_i$. Thus the derivative of *U* with respect of $\dot{x}_i$ will be zero. Also, from Equation (A.9), *T* is not a function of x_i. Thus its derivative with respect to x_i will be zero too. Thus, the right hand side of Equation (A.17) will be zero. Finally we obtain,

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{x}_i} - \frac{\partial L}{\partial x_i} = 0 \quad \textbf{(A.18)}$$

The importance of Lagrangian stems from the fact that once Lagrangian has been made, Equation (A.18) can be used to express the system just based on one of the variables.

The Lagrangian can also be expressed in terms of generalised coordinates and then the Lagrangian will be a function of the generalised coordinates i.e. $L(q, \dot{q}, t)$. Then, Equation (A.18) can be written as [11],

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_j} - \frac{\partial L}{\partial q_i} = 0 \quad \textbf{(A.19)}$$

This is called the *Lagrange* equation.

A.4 The Hamiltonian

We define *generalised momentum* as,

$$P_j = \frac{\partial L}{\partial \dot{q}_j} \quad \textbf{(A.20)}$$

Then, the *Lagrange* equation can be expressed as,

$$\frac{d}{dt}P_j = \frac{\partial L}{\partial q_j} = \frac{\partial T}{\partial q_j} - \frac{\partial U}{\partial q_j} \quad \textbf{(A.21)}$$

To see the variation of the Lagrangian with time we differentiate *L* with time t,

$$\frac{dL}{dt} = \sum_j \frac{\partial L}{\partial q_j}\frac{dq_j}{dt} + \sum_j \frac{\partial L}{\partial \dot{q}_j}\frac{d\dot{q}_j}{dt} + \frac{\partial L}{\partial t} \tag{A.22}$$

This equation can be rearranged as,

$$\frac{dL}{dt} = \frac{d}{dt}\Big(\sum_j \frac{\partial L}{\partial \dot{q}_j}\dot{q}_j\Big) + \frac{\partial L}{\partial t} \tag{A.23}$$

Now, if we define *Hamiltonian H* as,

$$H(q, \dot{q}, t) = \sum_j \dot{q}_j \frac{\partial L}{\partial \dot{q}_j} - L \tag{A.24}$$

Using the definition of *generalised momentum* from Equation (A.20) we can re-write the definition of *Hamiltonian* as,

$$H(q, \dot{q}, t) = \sum_j \dot{q}_j P_j - L \tag{A.25}$$

Using the value of Equation (A.25) in Equation (A.23) we obtain,

$$\frac{dH}{dt} = -\frac{\partial L}{\partial t} \tag{A.26}$$

If the *Lagrangian* does not depend on time, then *Hamiltonian* is also conserved.

A.5 Hamiltonian Equations

The variation of Lagrangian along a path can be expressed as,

$$\begin{aligned} dL &= \sum_j \frac{\partial L}{\partial q_j} dq_j + \sum_j \frac{\partial L}{\partial \dot{q}_j} \dot{q}_j + \frac{\partial L}{\partial t} dt \\ &= \sum_j \left(\dot{P}_j dq_j + P_j d\dot{q}_j\right) + \frac{\partial L}{\partial t} dt \end{aligned} \tag{A.27}$$

Then the change in *Hamiltonian* can be expressed as,

$$\begin{aligned} dH &= \sum_j \left(P_j d\dot{q}_j + \dot{q}_j dP_j\right) - dL \\ &= \sum_j \left(\dot{q}_j \, dP_j + \dot{P}_j \, dq_j\right) - dL \end{aligned} \tag{A.28}$$

Now the *Hamiltonian Equation of Motion* can be expressed as [11],

$$\dot{q}_j = \frac{\partial H}{\partial P_j}, \quad \dot{P}_J = -\frac{\partial H}{\partial q_j}, \quad \frac{\partial H}{\partial t} = -\frac{\partial L}{\partial t} \tag{A.29}$$

Appendix B

VARIATIONAL EQUATIONS: N BODY PROBLEM

The aim of this chapter is to formulate the various equations required by Section 2.6.1 for the variational equations. From equation (1.39) we know,

$$\ddot{\vec{r}_i} = \sum_{j=1, j\neq i}^{n} \frac{G\, m_j\, \vec{r_{ij}}}{r_{ij}^3}$$

where,

$$\vec{r_{ij}} = x_{ij}\ \hat{\imath} + y_{ij}\ \hat{\jmath} + z_{ij}\ \hat{k}$$

and,

$$x_{ij} = x_j - x_i \quad \textbf{(B.1)}$$

Thus, the three components of equation can be written as,

$$(\ddot{\vec{r}_i})_x = \sum_{j=1, j\neq i}^{n} \frac{G\, m_j\, x_{ij}}{r_{ij}^3} \quad \textbf{(B.2)}$$

$$(\ddot{\vec{r}_i})_y = \sum_{j=1, j\neq i}^{n} \frac{G\, m_j\, y_{ij}}{r_{ij}^3} \quad \textbf{(B.3)}$$

$$(\ddot{\vec{r}_i})_z = \sum_{j=1, j\neq i}^{n} \frac{G\, m_j\, z_{ij}}{r_{ij}^3} \quad \textbf{(B.4)}$$

Differentiating (B.2) with respect to x_i, y_i, and z_i we obtain
,

$$\frac{\partial(\ddot{\vec{r}_i})_x}{\partial x_i} = \sum_{j=1, j\neq i}^{n} G\, m_j \left[\frac{3\, x_{ij}^2}{r_{ij}^5} - \frac{1}{r_{ij}^3}\right] \quad \textbf{(B.5)}$$

$$\frac{\partial(\ddot{\vec{r}_i})_x}{\partial y_i} = \sum_{j=1, j\neq i}^{n} \frac{3\, G\, m_j\, x_{ij}\, y_{ij}}{r_{ij}^5} \quad \textbf{(B.6)}$$

$$\frac{\partial(\ddot{\vec{r}_i})_x}{\partial z_i} = \sum_{j=1, j\neq i}^{n} \frac{3\, G\, m_j\, x_{ij}\, z_{ij}}{r_{ij}^5} \quad \textbf{(B.7)}$$

Differentiating (B.3) with respect to x_i, y_i, and z_i we obtain,

$$\frac{\partial(\ddot{\vec{r}_i})_y}{\partial x_i} = \frac{\partial(\ddot{\vec{r}_i})_x}{\partial y_i} \quad \textbf{(B.8)}$$

$$\frac{\partial(\ddot{\vec{r}_i})_y}{\partial y_i} = \sum_{j=1, j\neq i}^{n} G\, m_j \left[\frac{3\, y_{ij}^2}{r_{ij}^5} - \frac{1}{r_{ij}^3}\right] \quad \textbf{(B.9)}$$

$$\frac{\partial(\ddot{\vec{r}}_i)_y}{\partial z_i} = \sum_{j=1, j\neq i}^{n} \frac{3\, G\, m_j\, y_{ij}\, z_{ij}}{r_{ij}^5} \tag{B.10}$$

Differentiating (B.4) with respect to x_i, y_i, and z_i we get,

$$\frac{\partial(\ddot{\vec{r}}_i)_z}{\partial x_i} = \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial z_i} \tag{B.11}$$

$$\frac{\partial(\ddot{\vec{r}}_i)_z}{\partial y_i} = \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial z_i} \tag{B.12}$$

$$\frac{\partial(\ddot{\vec{r}}_i)_z}{\partial z_i} = \sum_{j=1, j\neq i}^{n} G\, m_j \left[\frac{3\, z_{ij}^2}{r_{ij}^5} - \frac{1}{r_{ij}^3}\right] \tag{B.13}$$

Differentiating (B.2) with respect to *t* we get,

$$\frac{\partial(\ddot{\vec{r}}_i)_x}{\partial t} = \sum_{j=1, j\neq i}^{n} G\, m_j \left[\frac{\dot{x}_j}{r_{ij}^3} - \frac{3\, x_{ij}\,(x_{ij}\dot{x}_j + y_{ij}\dot{y}_j + z_{ij}\dot{z}_j)}{r_{ij}^5}\right] \tag{B.14}$$

Differentiating (B.3) with respect to *t* we get,

$$\frac{\partial(\ddot{\vec{r}}_i)_y}{\partial t} = \sum_{j=1, j\neq i}^{n} G\, m_j \left[\frac{\dot{y}_j}{r_{ij}^3} - \frac{3\, y_{ij}\,(x_{ij}\dot{x}_j + y_{ij}\dot{y}_j + z_{ij}\dot{z}_j)}{r_{ij}^5}\right] \tag{B.15}$$

Differentiating (B.4) with respect to *t* we get,

$$\frac{\partial(\ddot{\vec{r}}_i)_y}{\partial t} = \sum_{j=1, j\neq i}^{n} G\, m_j \left[\frac{\dot{z}_j}{r_{ij}^3} - \frac{3\, z_{ij}\,(x_{ij}\dot{x}_j + y_{ij}\dot{y}_j + z_{ij}\dot{z}_j)}{r_{ij}^5}\right] \tag{B.16}$$

Further calculation of the variational matrix can be seen in Section 2.6.1.

B.1 Variational Equations

The variational equations are given as,

$$\frac{d}{dt}\begin{pmatrix} X_7 \\ X_8 \\ X_9 \\ X_{10} \\ X_{11} \\ X_{12} \end{pmatrix} = \begin{pmatrix} X_{10} \\ X_{11} \\ X_{12} \\ \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial x_i} X_7 + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial y_i} X_8 + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial z_i} X_9 \\ \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial x_i} X_7 + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial y_i} X_8 + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial z_i} X_9 \\ \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial x_i} X_7 + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial y_i} X_8 + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial z_i} X_{9_s} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ V_t \end{pmatrix} \tag{B.17}$$

where,

$$V_t = \begin{pmatrix} \dfrac{\partial(\ddot{\vec{r}}_i)_x}{\partial t} \\ \dfrac{\partial(\ddot{\vec{r}}_i)_y}{\partial t} \\ \dfrac{\partial(\ddot{\vec{r}}_i)_z}{\partial t} \end{pmatrix} \quad \textbf{(B.18)}$$

and,

$$\begin{pmatrix} \dot{X}_{13} & \dot{X}_{19} & \cdots & \dot{X}_{43} \\ \dot{X}_{14} & \dot{X}_{20} & \cdots & \dot{X}_{44} \\ \vdots & \vdots & \vdots & \vdots \\ \dot{X}_{18} & \dot{X}_{24} & \cdots & \dot{X}_{48} \end{pmatrix} = \begin{pmatrix} X_{16} & X_{22} & \cdots & X_{46} \\ X_{17} & X_{23} & \cdots & X_{47} \\ X_{18} & X_{24} & \cdots & X_{48} \\ f_1 & f_4 & \cdots & f_{16} \\ f_2 & f_5 & \cdots & f_{17} \\ f_3 & f_6 & \cdots & f_{18} \end{pmatrix} \quad \textbf{(B.19)}$$

where,

$$f_1 = \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial x_i} X_{13} + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial y_i} X_{14} + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial z_i} X_{15} \quad \textbf{(B.20)}$$

$$f_2 = \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial x_i} X_{13} + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial y_i} X_{14} + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial z_i} X_{15} \quad \textbf{(B.21)}$$

$$f_3 = \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial x_i} X_{13} + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial y_i} X_{14} + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial z_i} X_{15} \quad \textbf{(B.22)}$$

$$f_4 = \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial x_i} X_{19} + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial y_i} X_{20} + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial z_i} X_{21} \quad \textbf{(B.23)}$$

$$f_5 = \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial x_i} X_{19} + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial y_i} X_{20} + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial z_i} X_{21} \quad \textbf{(B.24)}$$

$$f_6 = \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial x_i} X_{19} + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial y_i} X_{20} + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial z_i} X_{21} \quad \textbf{(B.25)}$$

$$f_7 = \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial x_i} X_{25} + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial y_i} X_{26} + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial z_i} X_{27} \quad \textbf{(B.26)}$$

$$f_8 = \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial x_i} X_{25} + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial y_i} X_{26} + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial z_i} X_{27} \quad \textbf{(B.27)}$$

$$f_9 = \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial x_i} X_{25} + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial y_i} X_{26} + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial z_i} X_{27} \quad \textbf{(B.28)}$$

$$f_{10} = \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial x_i} X_{31} + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial y_i} X_{32} + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial z_i} X_{33} \quad \textbf{(B.29)}$$

$$f_{11} = \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial x_i} X_{31} + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial y_i} X_{32} + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial z_i} X_{33} \quad \textbf{(B.30)}$$

$$f_{12} = \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial x_i} X_{31} + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial y_i} X_{32} + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial z_i} X_{33} \quad \textbf{(B.31)}$$

$$f_{13} = \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial x_i} X_{37} + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial y_i} X_{38} + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial z_i} X_{39} \quad \textbf{(B.32)}$$

$$f_{14} = \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial x_i} X_{37} + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial y_i} X_{38} + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial z_i} X_{39} \quad \textbf{(B.33)}$$

$$f_{15} = \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial x_i} X_{37} + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial y_i} X_{38} + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial z_i} X_{3}9 \quad \textbf{(B.34)}$$

$$f_{16} = \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial x_i} X_{43} + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial y_i} X_{44} + \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial z_i} X_{45} \quad \textbf{(B.35)}$$

$$f_{17} = \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial x_i} X_{43} + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial y_i} X_{44} + \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial z_i} X_{45} \quad \textbf{(B.36)}$$

$$f_{18} = \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial x_i} X_{43} + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial y_i} X_{44} + \frac{\partial(\ddot{\vec{r}}_i)_z}{\partial z_i} X_{45} \quad \textbf{(B.37)}$$

Appendix C

VARIATIONAL EQUATIONS: N-BODY W.R.T. A BODY IN SOLAR SYSTEM

C.1 Introduction

The aim of this chapter is to formulate the various equations required by Section 2.6.2 for the variational equations. From equation (1.45) we know,

$$\ddot{\vec{r_i}} = -\frac{G\,m_j\,\vec{r_i}}{r_i^3} + \sum_{k=1,k\neq i,j}^{n} G\,m_j\left[\frac{\vec{r_{ik}}}{r_{ik}^3} - \frac{\vec{r_k}}{r_k^3}\right]$$

Here, $\vec{r_i}$,$\vec{r_{ik}}$,$\vec{r_k}$ are vectors and can be expressed as,

$$\vec{r_i} = x_i\,\hat{\boldsymbol{\imath}} + y_i\,\hat{\boldsymbol{\jmath}} + z_i\,\hat{\boldsymbol{k}} \quad \textbf{(C.1)}$$

$$\vec{r_{ik}} = x_{ik}\,\hat{\boldsymbol{\imath}} + y_{ik}\,\hat{\boldsymbol{\jmath}} + z_{ik}\,\hat{\boldsymbol{k}} \quad \textbf{(C.2)}$$

$$\vec{r_k} = x_k\,\hat{\boldsymbol{\imath}} + y_k\,\hat{\boldsymbol{\jmath}} + z_k\,\hat{\boldsymbol{k}} \quad \textbf{(C.3)}$$

where,

$$x_{ik} = x_k - x_i \quad \textbf{(C.4)}$$

$$y_{ik} = y_k - y_i \quad \textbf{(C.5)}$$

$$z_{ik} = z_k - z_i \quad \textbf{(C.6)}$$

Thus the three components of equation can be written as,

$$(\ddot{\vec{r_i}})_x = -\frac{G\,m_j\,\vec{x_i}}{r_i^3} + \sum_{k=1,k\neq i,j}^{n} G\,m_j\left[\frac{\vec{x_{ik}}}{r_{ik}^3} - \frac{\vec{x_k}}{r_k^3}\right] \quad \textbf{(C.7)}$$

$$(\ddot{\vec{r_i}})_y = -\frac{G\,m_j\,\vec{y_i}}{r_i^3} + \sum_{k=1,k\neq i,j}^{n} G\,m_j\left[\frac{\vec{y_{ik}}}{r_{ik}^3} - \frac{\vec{y_k}}{r_k^3}\right] \quad \textbf{(C.8)}$$

$$(\ddot{\vec{r_i}})_z = -\frac{G\,m_j\,\vec{z_i}}{r_i^3} + \sum_{k=1,k\neq i,j}^{n} G\,m_j\left[\frac{\vec{z_{ik}}}{r_{ik}^3} - \frac{\vec{z_k}}{r_k^3}\right] \quad \textbf{(C.9)}$$

Differentiating equation (C.7) with respect to x_i, y_i, and z_i we obtain,

$$\frac{\partial(\ddot{\vec{r_i}})_x}{\partial x_i} = -G\,m_j\left[\frac{1}{r_i^3} - \frac{3\,x_i^2}{r_i^5}\right] + \sum_{k=1,k\neq i,j}^{n} G\,m_j\left[\frac{3\,x_{ik}^2}{r_{ik}^5} - \frac{1}{r_{ik}^3}\right] \quad \textbf{(C.10)}$$

$$\frac{\partial(\ddot{\vec{r_i}})_x}{\partial y_i} = G\,m_j\left[\frac{1}{r_i^3} - \frac{3\,x_i\,y_i}{r_i^5}\right] + \sum_{k=1,k\neq i,j}^{n} G\,m_j\left[\frac{3\,x_{ik}\,y_{ik}}{r_{ik}^5}\right] \quad \textbf{(C.11)}$$

$$\frac{\partial(\ddot{\vec{r}}_i)_x}{\partial z_i} = G\, m_j \left[\frac{1}{r_i^3} - \frac{3\, x_i\, z_i}{r_i^5} \right] + \sum_{k=1, k\neq i,j}^{n} G\, m_j \left[\frac{3\, x_{ik}\, z_{ik}}{r_{ik}^5} \right] \tag{C.12}$$

Differentiating equation (C.8) with respect to x_i, y_i, and z_i we obtain,

$$\frac{\partial(\ddot{\vec{r}}_i)_y}{\partial x_i} = \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial y_i} \tag{C.13}$$

$$\frac{\partial(\ddot{\vec{r}}_i)_y}{\partial y_i} = -G\, m_j \left[\frac{1}{r_i^3} - \frac{3\, y_i^2}{r_i^5} \right] + \sum_{k=1, k\neq i,j}^{n} G\, m_j \left[\frac{3\, y_{ik}^2}{r_{ik}^5} - \frac{1}{r_{ik}^3} \right] \tag{C.14}$$

$$\frac{\partial(\ddot{\vec{r}}_i)_y}{\partial z_i} = G\, m_j \left[\frac{1}{r_i^3} - \frac{3\, y_i\, z_i}{r_i^5} \right] + \sum_{k=1, k\neq i,j}^{n} G\, m_j \left[\frac{3\, y_{ik}\, z_{ik}}{r_{ik}^5} \right] \tag{C.15}$$

Differentiating equation (C.9) with respect to x_i, y_i, and z_i we obtain,

$$\frac{\partial(\ddot{\vec{r}}_i)_z}{\partial x_i} = \frac{\partial(\ddot{\vec{r}}_i)_x}{\partial z_i} \tag{C.16}$$

$$\frac{\partial(\ddot{\vec{r}}_i)_z}{\partial y_i} = \frac{\partial(\ddot{\vec{r}}_i)_y}{\partial z_i} \tag{C.17}$$

$$\frac{\partial(\ddot{\vec{r}}_i)_z}{\partial z_i} = -G\, m_j \left[\frac{1}{r_i^3} - \frac{3 z_i^2}{r_i^5} \right] + \sum_{k=1, k\neq i,j}^{n} G\, m_j \left[\frac{3\, z_{ik}^2}{r_{ik}^5} - \frac{1}{r_{ik}^3} \right] \tag{C.18}$$

Differentiating equation (C.7) with respect to *t* we obtain,

$$\begin{aligned}\frac{\partial(\ddot{\vec{r}}_i)_x}{\partial t} = \sum_{k=1, k\neq i,j}^{n} 3\, G\, m_j \Bigg[&- \frac{x_{ik}\,(x_{ik}\dot{x}_k + y_{ik}\dot{y}_k + z_{ik}\dot{z}_k)}{r_{ik}^5} \\ &+ \frac{x_k\,(x_k\dot{x}_k + y_k\dot{y}_k + z_k\dot{z}_k)}{r_k^5} \Bigg]\end{aligned} \tag{C.19}$$

Differentiating equation (C.8) with respect to *t* we obtain,

$$\begin{aligned}\frac{\partial(\ddot{\vec{r}}_i)_y}{\partial t} = \sum_{k=1, k\neq i,j}^{n} 3\, G\, m_j \Bigg[&- \frac{y_{ik}\,(x_{ik}\dot{x}_k + y_{ik}\dot{y}_k + z_{ik}\dot{z}_k)}{r_{ik}^5} \\ &+ \frac{y_k\,(x_k\dot{x}_k + y_k\dot{y}_k + z_k\dot{z}_k)}{r_k^5} \Bigg]\end{aligned} \tag{C.20}$$

Differentiating equation (C.9) with respect to t we obtain,

$$\frac{\partial(\ddot{\vec{r}}_i)_z}{\partial t} = \sum_{k=1,k\neq i,j}^{n} 3\,G\,m_j \Bigg[-\frac{z_{ik}\,(x_{ik}\dot{x}_k + y_{ik}\dot{y}_k + z_{ik}\dot{z}_k)}{r_{ik}^5} + \frac{z_k\,(x_k\dot{x}_k + y_k\dot{y}_k + z_k\dot{z}_k)}{r_k^5} \Bigg] \tag{C.21}$$

The variational matrix has been explained in Section 2.6.2. The variational equations are exactly similar to those in Section B.1.

Appendix D

VARIATIONAL EQUATIONS: QUASI PERIODIC FORMULATION OF N-BODY PROBLEM

This chapter formulates the Variational equations for Quasi-Periodic formulation of the restricted n-body problem. From Equations (1.141) to (1.143) we have,

$$\ddot{\vec{r}}_x = c_1 + c_4\dot{x} + c_5\dot{y} + c_7x + c_8y + c_9z + c_{13}\frac{\partial\Omega}{\partial x} \tag{D.1}$$

$$\ddot{\vec{r}}_y = c_2 - c_5\dot{x} + c_4\dot{y} + c_8x + c_{10}y + c_{11}z + c_{12}\frac{\partial\Omega}{\partial y} \tag{D.2}$$

$$\ddot{\vec{r}}_z = c_3 - c_6\dot{y} + c_4\dot{z} + c_9x - c_{11}y + c_{12}z + c_{13}\frac{\partial\Omega}{\partial z} \tag{D.3}$$

Differentiating Equation (D.1) with respect to x, y, z, $\dot{x}$, $\dot{y}$, and $\dot{z}$ we obtain,

$$\frac{\partial\ddot{\vec{r}}_x}{\partial x} = c_7 + c_{13}\Omega_{xx} \tag{D.4}$$

$$\frac{\partial\ddot{\vec{r}}_x}{\partial y} = c_8 + c_{13}\Omega_{xy} \tag{D.5}$$

$$\frac{\partial\ddot{\vec{r}}_x}{\partial z} = c_9 + c_{13}\Omega_{xz} \tag{D.6}$$

$$\frac{\partial\ddot{\vec{r}}_x}{\partial\dot{x}} = c_4 \tag{D.7}$$

$$\frac{\partial\ddot{\vec{r}}_x}{\partial\dot{y}} = c_5 \tag{D.8}$$

$$\frac{\partial\ddot{\vec{r}}_x}{\partial\dot{z}} = 0 \tag{D.9}$$

Furthermore, differentiating Equation (D.2) with respect to x, y, z, $\dot{x}$, $\dot{y}$, and $\dot{z}$ we obtain,

$$\frac{\partial\ddot{\vec{r}}_y}{\partial x} = -c_8 + c_{13}\Omega_{yx} \tag{D.10}$$

$$\frac{\partial\ddot{\vec{r}}_y}{\partial y} = c_{10} + c_{13}\Omega_{yy} \tag{D.11}$$

$$\frac{\partial\ddot{\vec{r}}_y}{\partial z} = c_{11} + c_{13}\Omega_{yz} \tag{D.12}$$

$$\frac{\partial\ddot{\vec{r}}_y}{\partial\dot{x}} = -c_5 \tag{D.13}$$

$$\frac{\partial \ddot{\vec{r}}_y}{\partial \dot{y}} = c_4 \tag{D.14}$$

$$\frac{\partial \ddot{\vec{r}}_y}{\partial \dot{z}} = c_6 \tag{D.15}$$

Also, differentiating Equation (D.3) with respect to x, y, z, $\dot{x}$, $\dot{y}$, and $\dot{z}$ we obtain,

$$\frac{\partial \ddot{\vec{r}}_z}{\partial x} = c_9 + c_{13}\Omega_{zx} \tag{D.16}$$

$$\frac{\partial \ddot{\vec{r}}_z}{\partial y} = -c_{11} + c_{13}\Omega_{zy} \tag{D.17}$$

$$\frac{\partial \ddot{\vec{r}}_z}{\partial z} = c_{12} + c_{13}\Omega_{zz} \tag{D.18}$$

$$\frac{\partial \ddot{\vec{r}}_z}{\partial \dot{x}} = 0 \tag{D.19}$$

$$\frac{\partial \ddot{\vec{r}}_z}{\partial \dot{y}} = -c_6 \tag{D.20}$$

$$\frac{\partial \ddot{\vec{r}}_z}{\partial \dot{z}} = c_4 \tag{D.21}$$

Further, differentiating Equations (D.1) to (D.3) with respect to *t* we get,

$$\frac{\partial \ddot{\vec{r}}_x}{\partial t} = \dot{c_1} + \dot{c_4}\dot{x} + \dot{c_5}\dot{y} + \dot{c_7}x + \dot{c_8}y + \dot{c_9}z + \dot{c_{13}}\Omega_x + c_{13}\Omega_{xt} \tag{D.22}$$

$$\frac{\partial \ddot{\vec{r}}_y}{\partial t} = \dot{c_2} - \dot{c_5}\dot{y} + \dot{c_4}\dot{y} + \dot{c_6}\dot{z} - \dot{c_8}x + \dot{c_{10}}y + \dot{c_{11}}z + \dot{c_{13}}\Omega_y + c_{13}\Omega_{yt} \tag{D.23}$$

$$\frac{\partial \ddot{\vec{r}}_z}{\partial t} = \dot{c_3} - \dot{c_6}\dot{y} + \dot{c_4}\dot{z} + \dot{c_9}x - \dot{c_{11}}y + \dot{c_{12}}z + \dot{c_{13}}\Omega_z + c_{13}\Omega_{zt} \tag{D.24}$$

For the equations above we require the differential of the coefficients of the quasi-periodic formulation of the restricted three body problem. Differentiating Equations (1.147) to (1.159) with respect to time we obtain,

$$\dot{c_1} = \frac{(\ddot{\vec{B}}\,.\,C_1)\,\dot{k} - k\,(\ddot{\vec{B}}\,.\,\dot{C}_1 + \dddot{\vec{B}}\,.\,C_1)}{k^2} \tag{D.25}$$

$$\dot{c_2} = \frac{(\ddot{\vec{B}}\,.\,C_2)\,\dot{k} - k\,(\ddot{\vec{B}}\,.\,\dot{C}_2 + \dddot{\vec{B}}\,.\,C_2)}{k^2} \tag{D.26}$$

$$\dot{c_3} = \frac{(\ddot{\vec{B}}\,.\,C_3)\,\dot{k} - k\,(\ddot{\vec{B}}\,.\,\dot{C}_3 + \dddot{\vec{B}}\,.\,C_3)}{k^2} \tag{D.27}$$

$$\dot{c_4} = 2\left(\frac{\dot{k}^2}{k^2} - \frac{\ddot{k}}{k}\right) \tag{D.28}$$

$$\dot{c_5} = 2\,(\dot{C}_1\,.\,\dot{C}_2 + \ddot{C}_1\,.\,C_2) \tag{D.29}$$

$$\dot{c_6} = 2\,(\dot{C}_2\,.\,\dot{C}_3 + \ddot{C}_2\,.\,C_3) \tag{D.30}$$

$$\dot{c_7} = 2\,(\dot{C_1}\,.\,\ddot{C_1}) - \left(\frac{k\dddot{k} - \ddot{k}\dot{k}}{k^2}\right) \tag{D.31}$$

$$\dot{c_8} = 2\left(\frac{\ddot{k}k - \dot{k}^2}{k^2}\right)(\dot{C_1}\,.\,C_2) + 2\,\frac{\dot{k}}{k}(\dot{C_1}\,.\,\dot{C_2} + \ddot{C_1}\,.\,C_2) + \ddot{C_1}\,.\,\dot{C_2} + \dddot{C_1}\,.\,C_2 \tag{D.32}$$

$$\dot{c_9} = \dot{C_1}\,.\,\ddot{C_3} + \ddot{C_1}\,.\,\dot{C_3} \tag{D.33}$$

$$\dot{c_{10}} = 2\,(\dot{C_2}\,.\,\ddot{C_2}) - \frac{\dddot{k}\,k - \ddot{k}\dot{k}}{k^2} \tag{D.34}$$

$$\dot{c_{11}} = 2\left(\frac{\ddot{k}k - \dot{k}^2}{k^2}\right)(\dot{C_2}\,.\,C_3) + 2\,\frac{\dot{k}}{k}(\dot{C_2}\,.\,\dot{C_3} + \ddot{C_2}\,.\,C_3) + \ddot{C_2}\,.\,\dot{C_3} + \dddot{C_2}\,.\,C_3 \tag{D.35}$$

$$\dot{c_{12}} = 2\,(\dot{C_3}\,.\,\ddot{C_3}) + \left(\frac{\dddot{k}\,k - \dot{k}\ddot{k}}{k^2}\right) \tag{D.36}$$

$$\dot{c_{13}} = -\frac{3\,a^3\,\dot{k}}{k^2} \tag{D.37}$$

The computation of $\dddot{C}_1$, $\dddot{C}_2$, and $\dddot{C}_3$ are shown in Appendix E.

Also, as described by Equation (1.144) we have,

$$\Omega(x,y,z) = \frac{1-\mu}{\sqrt{(x-\mu)^2 + y^2 + z^2}} + \frac{\mu}{\sqrt{(x-\mu+1)^2 + y^2 + z^2}} + \sum_{A\epsilon(S,E,M,P...P_k),A\neq m_1,m_2} \frac{\mu_A}{\|\vec{r} - \vec{r_A}\|}$$

Differentiating this equation with respect to *x* we get,

$$\begin{aligned}\frac{\partial\Omega}{\partial x} = &-(1-\mu)\left[\left((x-\mu)^2 + y^2 + z^2\right)^{-3/2}(x-\mu)\right] - \mu\left[\left((x-\mu+1)^2 + y^2\right.\right.\\ &\left.\left.+z^2\right)^{-3/2}(x-\mu+1)\right] - \sum_{A\epsilon(S,E,M,P...,P_k),A\neq m_1,m_2} \mu_A\left[\left((x-x_A)^2 + (y-y_A)^2\right.\right.\\ &\left.\left.+(z-z_A)^2\right)^{-3/2}(x-x_A)\right]\end{aligned} \tag{D.38}$$

Differentiating the Equation (1.144) with respect to *y* we get,

$$\begin{aligned}\frac{\partial\Omega}{\partial y} = &-(1-\mu)\left[\left((x-\mu)^2 + y^2 + z^2\right)^{-3/2}y\right] - \mu\left[\left((x-\mu+1)^2 + y^2 + z^2\right)^{-3/2}y\right]\\ &- \sum_{A\epsilon(S,E,M,P...,P_k),A\neq m_1,m_2} \mu_A\left[\left((x-x_A)^2 + (y-y_A)^2 + (z-z_A)^2\right)^{-3/2}(y-y_A)\right]\end{aligned} \tag{D.39}$$

Differentiating the Equation (1.144) with respect to *z* we get,

$$\begin{aligned}\frac{\partial \Omega}{\partial z} &= -(1-\mu)\left[\left((x-\mu)^2+y^2+z^2\right)^{-3/2} z\right] - \mu\left[\left((x-\mu+1)^2+y^2+z^2\right)^{-3/2} z\right] \\ &- \sum_{A\epsilon(S,E,M,P...,P_k),A\neq m_1,m_2} \mu_A \left[\left((x-x_A)^2+(y-y_A)^2+(z-z_A)^2\right)^{-3/2}(z-z_A)\right]\end{aligned} \tag{D.40}$$

Differentiating Equation (D.38) with respect to *x*, *y*, and *z* we get,

$$\begin{aligned}\frac{\partial^2 \Omega}{\partial^2 x^2} &= -(1-\mu)\left[\left((x-\mu)^2+y^2+z^2\right)^{-3/2} - 3\left((x-\mu)^2+y^2+z^2\right)^{-5/2}(x-\mu)^2\right] \\ &-\mu\left[\left((x-\mu+1)^2+y^2+z^2\right)^{-3/2} - 3\left((x-\mu+1)^2+y^2+z^2\right)^{-5/2}(x-\mu)^2\right] \\ &- \sum_{A\epsilon(S,E,M,P...,P_k),A\neq m_1,m_2} \mu_A \left[\left((x-x_A)^2+(y-y_A)^2+(z-z_A)^2\right)^{-3/2} - 3\left((x-x_A)^2\right.\right. \\ &\left.\left. +(y-y_A)^2+(z-z_A)^2\right)^{-5/2}(x-x_A)^2\right]\end{aligned} \tag{D.41}$$

and,

$$\begin{aligned}\frac{\partial^2 \Omega}{\partial x \partial y} &= 3\,(1-\mu)\left[y\left((x-\mu)^2+y^2+z^2\right)^{-5/2}[x-\mu]\right] + 3\,\mu\left[y\left((x-\mu+1)^2+y^2+z^2\right)^{-5/2}\right. \\ &\left.(x-\mu+1)\right] + 3\sum_{A\epsilon(S,E,M,P...,P_k),A\neq m_1,m_2} \mu_A \left[\left((x-x_A)^2+(y-y_A)^2+(z-z_A)^2\right)^{-5/2}\right. \\ &\left.(y-y_A)(x-x_A)\right]\end{aligned} \tag{D.42}$$

and,

$$\begin{aligned}\frac{\partial^2 \Omega}{\partial x \partial z} &= 3\,(1-\mu)\left[z\left((x-\mu)^2+y^2+z^2\right)^{-5/2}(x-\mu)\right] + 3\,\mu\left[z\left((x-\mu+1)^2+y^2+z^2\right)^{-5/2}\right. \\ &\left.(x-\mu+1)\right] + 3\sum_{A\epsilon(S,E,M,P...,P_k),A\neq m_1,m_2} \mu_A \left[\left((x-x_A)^2+(y-y_A)^2+(z-z_A)^2\right)^{-5/2}\right. \\ &\left.(z-z_A)\,(x-x_A)\right]\end{aligned} \tag{D.43}$$

Further, differentiating Equation (D.39) with respect to *x*, *y*, and *z* we get,

$$
\begin{aligned}
\frac{\partial^2\Omega}{\partial^2 y^2} &= -(1-\mu)\left[\left((x-\mu)^2+y^2+z^2\right)^{-3/2} - 3\left((x-\mu)^2+y^2+z^2]^{-5/2}y^2\right)\right] \\
&\quad -\mu\left[\left((x-\mu+1)^2+y^2+z^2\right)^{-3/2} - 3\left((x-\mu+1)^2+y^2+z^2\right)^{-5/2}y^2\right] \\
&\quad - \sum_{A\epsilon(S,E,M,P\ldots,P_k),A\neq m_1,m_2} \mu_A\left[\left((x-x_A)^2+(y-y_A)^2+(z-z_A)^2\right)^{-3/2} - 3\Big((x-x_A)^2+\right. \\
&\quad \left.(y-y_A)^2+(z-z_A)^2\Big)^{-5/2}(y-y_A)^2\right]
\end{aligned}
\tag{D.44}
$$

$$
\frac{\partial^2\Omega}{\partial y\partial x} = \frac{\partial^2\Omega}{\partial x\partial x}
\tag{D.45}
$$

$$
\begin{aligned}
\frac{\partial^2\Omega}{\partial y\partial z} &= 3\,(1-\mu)\left[z\Big((x-\mu)^2+y^2+z^2\Big)^{-5/2}y + 3\,\mu\, z\left[\left((x-\mu+1)^2+y^2+z^2\right)^{-5/2}y\right]\right. \\
&\quad +3\sum_{A\epsilon(S,E,M,P\ldots,P_k),A\neq m_1,m_2} \mu_A\left[\left((x-x_A)^2+(y-y_A)^2+(z-z_A)^2\right)^{-5/2}\right. \\
&\quad \left.(z-z_A)\,(y-y_A)\right]
\end{aligned}
\tag{D.46}
$$

Differentiating Equation (D.40) with respect to *x*, *y*, and *z* we get,

$$
\frac{\partial^2\Omega}{\partial z\partial x} = \frac{\partial^2\Omega}{\partial x\partial z}
\tag{D.47}
$$

$$
\frac{\partial^2\Omega}{\partial z\partial y} = \frac{\partial^2\Omega}{\partial y\partial z}
\tag{D.48}
$$

$$
\begin{aligned}
\frac{\partial^2\Omega}{\partial^2 z^2} &= -(1-\mu)\left[\left((x-\mu)^2+y^2+z^2\right)^{-3/2} - 3\left((x-\mu)^2+y^2+z^2\right)^{-5/2}z^2\right] \\
&\quad -\mu\left[\left((x-\mu+1)^2+y^2+z^2\right)^{-3/2} - 3\left((x-\mu+1)^2+y^2+z^2\right)^{-5/2}z^2\right] \\
&\quad - \sum_{A\epsilon(S,E,M,P\ldots,P_k),A\neq m_1,m_2} \mu_A\left[\left((x-x_A)^2+(y-y_A)^2+(z-z_A)^2\right)^{-3/2} - 3\left([x-x_A)^2\right.\right. \\
&\quad \left.\left.+(y-y_A)^2+(z-z_A)^2\right)^{-5/2}(z-z_A)^2\right]
\end{aligned}
\tag{D.49}
$$

To find the dependence of time we need to differentiate Equation (D.38) with respect to *t*. Doing this we get,

$$\frac{\partial^2\Omega}{\partial x\partial t} = \sum_{A\epsilon(S,E,M,P...,P_k),A\neq m_1,m_2} \mu_A \left((x-x_A)^2+(y-y_A)^2+(z-z_A)^2\right)^{-3/2}\dot{x}_A$$
$$-3\,(x-x_A)\left[\left((x-x_A)^2+(y-y_A)^2+(z-z_A)^2\right)^{-5/2}\left((x-x_A)\,\dot{x_A}\right.\right.$$
$$\left.\left.+(y-y_A)\,\dot{y_A}+(z-z_A)\,\dot{z_A}\right)\right] \tag{D.50}$$

Differentiating Equation (D.39) with respect to *t* we obtain,

$$\frac{\partial^2\Omega}{\partial y\partial t} = \sum_{A\epsilon(S,E,M,P...,P_k),A\neq m_1,m_2} \mu_A \left((x-x_A)^2+(y-y_A)^2+(z-z_A)^2\right)^{-3/2}\dot{y}_A$$
$$-3\,(y-y_A)\left[\left((x-x_A)^2+(y-y_A)^2+(z-z_A)^2\right)^{-5/2}\left((x-x_A)\,\dot{x_A}\right.\right.$$
$$\left.\left.+(y-y_A)\,\dot{y_A}+(z-z_A)\,\dot{z_A}\right)\right] \tag{D.51}$$

Further, differentiating Equation (D.39) with respect to *t* we get,

$$\frac{\partial^2\Omega}{\partial z\partial t} = \sum_{A\epsilon(S,E,M,P...,P_k),A\neq m_1,m_2} \mu_A \left((x-x_A)^2+(y-y_A)^2+(z-z_A)^2\right)^{-3/2}\dot{z}_A$$
$$-3(z-z_A)\left[\left((x-x_A)^2+(y-y_A)^2+(z-z_A)^2\right)^{-5/2}\left((x-x_A)\,\dot{x_A}\right.\right.$$
$$\left.\left.+(y-y_A)\,\dot{y_A}+(z-z_A)\,\dot{z_A}\right)\right] \tag{D.52}$$

D.1 Variational Equations

The variational equations are given as,

$$\begin{pmatrix} \dot{X}_7 \\ \dot{X}_8 \\ \dot{X}_9 \\ \dot{X}_{10} \\ \dot{X}_{11} \\ \dot{X}_{12} \end{pmatrix} = \begin{pmatrix} \dot{X}_{10} \\ \dot{X}_{11} \\ \dot{X}_{12} \\ \frac{\partial\ddot{\vec{r}}_x}{\partial x}X_7 + \frac{\partial\ddot{\vec{r}}_x}{\partial y}X_8 + \frac{\partial\ddot{\vec{r}}_x}{\partial z}X_9 + \frac{\partial\ddot{\vec{r}}_x}{\partial \dot{x}}X_{10} + \frac{\partial\ddot{\vec{r}}_x}{\partial \dot{y}}X_{11} + \frac{\partial\ddot{\vec{r}}_x}{\partial \dot{z}}X_{12} \\ \frac{\partial\ddot{\vec{r}}_y}{\partial x}X_7 + \frac{\partial\ddot{\vec{r}}_y}{\partial y}X_8 + \frac{\partial\ddot{\vec{r}}_y}{\partial z}X_9 + \frac{\partial\ddot{\vec{r}}_y}{\partial \dot{x}}X_{10} + \frac{\partial\ddot{\vec{r}}_y}{\partial \dot{y}}X_{11} + \frac{\partial\ddot{\vec{r}}_y}{\partial \dot{z}}X_{12} \\ \frac{\partial\ddot{\vec{r}}_z}{\partial x}X_7 + \frac{\partial\ddot{\vec{r}}_z}{\partial y}X_8 + \frac{\partial\ddot{\vec{r}}_z}{\partial z}X_9 + \frac{\partial\ddot{\vec{r}}_z}{\partial \dot{x}}X_{10} + \frac{\partial\ddot{\vec{r}}_z}{\partial \dot{y}}X_{11} + \frac{\partial\ddot{\vec{r}}_z}{\partial \dot{z}}X_{12} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ V_t \end{pmatrix} \tag{D.53}$$

where,

$$V_t = \begin{pmatrix} \dfrac{\partial \ddot{\vec{r}}_x}{\partial t} \\ \dfrac{\partial \ddot{\vec{r}}_y}{\partial t} \\ \dfrac{\partial \ddot{\vec{r}}_z}{\partial t} \end{pmatrix} \tag{D.54}$$

and,

$$\begin{pmatrix} \dot{X}_{13} & \dot{X}_{19} & \cdots & \dot{X}_{43} \\ \dot{X}_{14} & \dot{X}_{20} & \cdots & \dot{X}_{44} \\ \vdots & \vdots & \vdots & \vdots \\ \dot{X}_{18} & \dot{X}_{24} & \cdots & \dot{X}_{48} \end{pmatrix} = \begin{pmatrix} X_{16} & X_{22} & \cdots & X_{46} \\ X_{17} & X_{23} & \cdots & X_{47} \\ X_{18} & X_{24} & \cdots & X_{48} \\ f_1 & f_4 & \cdots & f_{16} \\ f_2 & f_5 & \cdots & f_{17} \\ f_3 & f_6 & \cdots & f_{18} \end{pmatrix} \tag{D.55}$$

where,

$$f_1 = \frac{\partial \ddot{\vec{r}}_x}{\partial x} X_{13} + \frac{\partial \ddot{\vec{r}}_x}{\partial y} X_{14} + \frac{\partial \ddot{\vec{r}}_x}{\partial z} X_{15} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{x}} X_{16} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{y}} X_{17} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{z}} X_{18} \tag{D.56}$$

$$f_2 = \frac{\partial \ddot{\vec{r}}_y}{\partial x} X_{13} + \frac{\partial \ddot{\vec{r}}_y}{\partial y} X_{14} + \frac{\partial \ddot{\vec{r}}_y}{\partial z} X_{15} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{x}} X_{16} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{y}} X_{17} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{z}} X_{18} \tag{D.57}$$

$$f_3 = \frac{\partial \ddot{\vec{r}}_z}{\partial x} X_{13} + \frac{\partial \ddot{\vec{r}}_z}{\partial y} X_{14} + \frac{\partial \ddot{\vec{r}}_z}{\partial z} X_{15} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{x}} X_{16} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{y}} X_{17} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{z}} X_{18} \tag{D.58}$$

$$f_4 = \frac{\partial \ddot{\vec{r}}_x}{\partial x} X_{19} + \frac{\partial \ddot{\vec{r}}_x}{\partial y} X_{20} + \frac{\partial \ddot{\vec{r}}_x}{\partial z} X_{21} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{x}} X_{22} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{y}} X_{23} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{z}} X_{24} \tag{D.59}$$

$$f_5 = \frac{\partial \ddot{\vec{r}}_y}{\partial x} X_{19} + \frac{\partial \ddot{\vec{r}}_y}{\partial y} X_{20} + \frac{\partial \ddot{\vec{r}}_y}{\partial z} X_{21} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{x}} X_{22} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{y}} X_{23} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{z}} X_{24} \tag{D.60}$$

$$f_6 = \frac{\partial \ddot{\vec{r}}_z}{\partial x} X_{19} + \frac{\partial \ddot{\vec{r}}_z}{\partial y} X_{20} + \frac{\partial \ddot{\vec{r}}_z}{\partial z} X_{21} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{x}} X_{22} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{y}} X_{23} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{z}} X_{24} \tag{D.61}$$

$$f_7 = \frac{\partial \ddot{\vec{r}}_x}{\partial x} X_{25} + \frac{\partial \ddot{\vec{r}}_x}{\partial y} X_{26} + \frac{\partial \ddot{\vec{r}}_x}{\partial z} X_{27} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{x}} X_{28} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{y}} X_{29} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{z}} X_{30} \tag{D.62}$$

$$f_8 = \frac{\partial \ddot{\vec{r}}_y}{\partial x} X_{25} + \frac{\partial \ddot{\vec{r}}_y}{\partial y} X_{26} + \frac{\partial \ddot{\vec{r}}_y}{\partial z} X_{27} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{x}} X_{28} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{y}} X_{29} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{z}} X_{30} \tag{D.63}$$

$$f_9 = \frac{\partial \ddot{\vec{r}}_z}{\partial x} X_{25} + \frac{\partial \ddot{\vec{r}}_z}{\partial y} X_{26} + \frac{\partial \ddot{\vec{r}}_z}{\partial z} X_{27} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{x}} X_{28} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{y}} X_{29} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{z}} X_{30} \tag{D.64}$$

$$f_{10} = \frac{\partial \ddot{\vec{r}}_x}{\partial x} X_{31} + \frac{\partial \ddot{\vec{r}}_x}{\partial y} X_{32} + \frac{\partial \ddot{\vec{r}}_x}{\partial z} X_{33} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{x}} X_{34} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{y}} X_{35} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{z}} X_{36} \tag{D.65}$$

$$f_{11} = \frac{\partial \ddot{\vec{r}}_y}{\partial x} X_{31} + \frac{\partial \ddot{\vec{r}}_y}{\partial y} X_{32} + \frac{\partial \ddot{\vec{r}}_y}{\partial z} X_{33} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{x}} X_{34} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{y}} X_{35} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{z}} X_{36} \tag{D.66}$$

$$f_{12} = \frac{\partial \ddot{\vec{r}}_z}{\partial x} X_{31} + \frac{\partial \ddot{\vec{r}}_z}{\partial y} X_{32} + \frac{\partial \ddot{\vec{r}}_z}{\partial z} X_{33} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{x}} X_{34} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{y}} X_{35} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{z}} X_{36} \tag{D.67}$$

$$f_{13} = \frac{\partial \ddot{\vec{r}}_x}{\partial x} X_{37} + \frac{\partial \ddot{\vec{r}}_x}{\partial y} X_{38} + \frac{\partial \ddot{\vec{r}}_x}{\partial z} X_{39} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{x}} X_{40} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{y}} X_{41} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{z}} X_{42} \quad \textbf{(D.68)}$$

$$f_{14} = \frac{\partial \ddot{\vec{r}}_y}{\partial x} X_{37} + \frac{\partial \ddot{\vec{r}}_y}{\partial y} X_{38} + \frac{\partial \ddot{\vec{r}}_y}{\partial z} X_{39} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{x}} X_{40} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{y}} X_{41} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{z}} X_{42} \quad \textbf{(D.69)}$$

$$f_{15} = \frac{\partial \ddot{\vec{r}}_z}{\partial x} X_{37} + \frac{\partial \ddot{\vec{r}}_z}{\partial y} X_{38} + \frac{\partial \ddot{\vec{r}}_z}{\partial z} X_{39} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{x}} X_{40} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{y}} X_{41} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{z}} X_{42} \quad \textbf{(D.70)}$$

$$f_{16} = \frac{\partial \ddot{\vec{r}}_x}{\partial x} X_{43} + \frac{\partial \ddot{\vec{r}}_x}{\partial y} X_{44} + \frac{\partial \ddot{\vec{r}}_x}{\partial z} X_{45} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{x}} X_{46} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{y}} X_{47} + \frac{\partial \ddot{\vec{r}}_x}{\partial \dot{z}} X_{48} \quad \textbf{(D.71)}$$

$$f_{17} = \frac{\partial \ddot{\vec{r}}_y}{\partial x} X_{43} + \frac{\partial \ddot{\vec{r}}_y}{\partial y} X_{44} + \frac{\partial \ddot{\vec{r}}_y}{\partial z} X_{45} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{x}} X_{46} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{y}} X_{47} + \frac{\partial \ddot{\vec{r}}_y}{\partial \dot{z}} X_{48} \quad \textbf{(D.72)}$$

$$f_{18} = \frac{\partial \ddot{\vec{r}}_z}{\partial x} X_{43} + \frac{\partial \ddot{\vec{r}}_z}{\partial y} X_{44} + \frac{\partial \ddot{\vec{r}}_z}{\partial z} X_{45} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{x}} X_{46} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{y}} X_{47} + \frac{\partial \ddot{\vec{r}}_z}{\partial \dot{z}} X_{48} \quad \textbf{(D.73)}$$

Appendix E

THIRD DERIVATIVE OF MATRIX C

The aim of this chapter is to present the values of the third derivative of the matrix *C* from Equation (1.10) with respect to time. The matrix *C'''* will be given by,

$$C''' = \left(C_1''' \quad C_2''' \quad C_3'''\right) \tag{E.1}$$

Here, C_1''' can be obtained by differentiating Equation (1.33) with respect to t. It can be expressed as,

$$\begin{aligned} C_1''' = & \frac{k'\,\vec{R_{21}}'' - \vec{R_{21}}'''\,k}{k^2} + 2\left(\frac{k^2\,k''\,\vec{R_{21}}' + k^2\,k'\,\vec{R_{21}}'' - 2\,k\,k'^2\,\vec{R_{21}}'}{k^4}\right) \\ & + \vec{R_{21}}'\left(\frac{k''\,k - 2\,k'^2}{k\,\|\vec{R_{21}}\|_2^2}\right) + \vec{R_{21}}\left(\frac{k'''}{k^2} - \frac{k''\,(4\,k' + 3)}{k^3} - \frac{6\,k'^2}{k^4}\right) \end{aligned} \tag{E.2}$$

C_3''' can be obtained by differentiating Equation (1.34) with respect to time. It can be expressed as,

$$C_3''' = \vec{e_f}' - \left(2\,(C_3' \,.\, C_3'') + C_3 \,.\, \vec{e_F}' + C_3' \,.\, \vec{e_F}\right) C_3 + \left(C_3' \,.\, C_3' + C_3 \,.\vec{e_F}\right) C_3 \tag{E.3}$$

where, $\vec{e_F}$ is given by Equation (1.36). Also, $\vec{e_F}'$ can be obtained by Equation (1.36) with respect to time as,

$$\begin{aligned} e_F' = & \frac{1}{\|\vec{R_{21}} \times \vec{R_{21}}'\|_2}\Big[\vec{R_{21}}' \times \vec{R_{21}}''' + \vec{R_{21}}' \times \vec{R_{21}}''' + \vec{R_{21}} \times \vec{R_{21}}'''' \\ & -2\left(R_L'\,(C_1 \times C_1'') + (\vec{R_{21}}' \times \vec{R_{21}}'' + \vec{R_{21}} \times \vec{R_{21}}''')\,R_L\right)\Big] \\ & - \frac{1}{\|\vec{R_{21}} \times \vec{R_{21}}'\|_2}\Big[C_3 \,.\, (\vec{R_{21}} \times \vec{R_{21}}'')\left(\vec{R_{21}}' \times \vec{R_{21}}'' + \vec{R_{21}} \times \vec{R_{21}}'''\right. \\ & \left. -2\,R_L\,(\vec{R_{21}} \times \vec{R_{21}}'')\right)\Big] \end{aligned} \tag{E.4}$$

In this equation, R_L is given by,

$$R_L = \frac{C_3 \,.\, (\vec{R_{21}} \times \vec{R_{21}}'')}{\|\vec{R_{21}} \times \vec{R_{21}}'\|_2} \tag{E.5}$$

Differentiating this equation with respect to time we can get R_L' as,

$$R_L' = \frac{1}{\|\vec{R_{21}} \times \vec{R_{21}}'\|_2}\Big[C_3 \,.\, (\vec{R_{21}}' \times \vec{R_{21}}'' + \vec{R_{21}} \times \vec{R_{21}}''') \\ +C_3' \,.\, (\vec{R_{21}} \times \vec{R_{21}}'')\Big] - R_L^2 \tag{E.6}$$

In these equations k''' can be obtained by differentiating Equation (1.37) with respect to time as,

$$k''' = \frac{3\ (\vec{R_{21}}'' \,.\, \vec{R_{21}}' - k'\,k'') + \vec{R_{21}} \,.\, \vec{R_{21}}'''}{k} \tag{E.7}$$

C_2''' can be obtained by differentiating Equation (1.35) with respect to time as,

$$C_2''' = C_3''' \times C_1 + C_3 \times C_1''' + 3\,(C_3'' \times C_1' + C_3' \times C_1'') \tag{E.8}$$

Appendix F

MODELEPH.DAT FILE

```
'../../ephemerides/'
1 1 DE406s 0 !Mercury barycenter DE406s
2 1 DE406s 3 !Venus barycenter
3 1 DE406s 2 !Earth barycenter !Take the Earth respect SSB
4 1 DE406s 0 !Mars barycenter
5 1 DE406s 1 !Jupiter barycenter
6 1 DE406s 0 !Saturn barycenter
7 1 DE406s 0 !Uranus barycenter
8 1 DE406s 0 !Neptune barycenter
9 1 DE406s 0 !Pluto barycenter
301 1 DE406s !Moon
10 1 DE406s !Sun
199 1 DE406s !Mercury
299 1 DE406s !Venus
399 1 DE406s !Earth
499 0 DE406s !Mars
401 0 MAR085 !Phobos MAR085
402 0 MAR085 !Deimos
499 1 MAR085 !Mars
501 1 JUP230L !Io JUP230L
502 1 JUP230L !Europa
503 0 JUP230L !Ganymede
504 0 JUP230L !Callisto
505 0 JUP230L !Amalthea
514 0 JUP230L !Thebe
599 1 JUP230L !Jupiter
601 0 SAT317 !Mimas SAT317
602 0 SAT317 !Enceladus
603 0 SAT317 !Tethys
604 0 SAT317 !Dione
605 0 SAT317 !Rhea
606 1 SAT317 !Titan
607 0 SAT317 !Hyperion
608 0 SAT317 !Iapetus
609 0 SAT317 !Phoebe
612 0 SAT317 !Helene
613 0 SAT317 !Telesto
614 0 SAT317 !Calypso
632 0 SAT317 !Methon
699 1 SAT317 !Saturn
701 0 URA083 !Ariel URA083
```

```
702 0 URA083 !Umbriel
703 0 URA083 !Titania
704 0 URA083 !Oberon
705 0 URA083 !Miranda
799 1 URA083 !Uranus
801 0 NEP081 !Triton NEP081
802 0 NEP081 !Nereid
808 0 NEP081 !Proteus
899 1 NEP081 !Neptune
901 0 PLU017XL !Charon PLU017XL
902 0 PLU017XL !Nix
903 0 PLU017XL !Hydra
999 1 PLU017XL !Pluto
```

Appendix G

BODIES IN JPL EPHEMERIS DE406

This chapter lists the bodies in the JPL Ephemeris that will be used in this thesis.

G.1 Sun

Table G.1 Bodies in Sun

Body	Identifier
Sun	10
Solar System Barycenter	0

G.2 The Mercurian System

Table G.2 Bodies in Mercurian System

Body	Identifier
Mercury Barycenter	1
Mercury	199

G.3 The Venusian System

Table G.3 Bodies in Venusian System

Body	Identifier
Venus Barycenter	2
Venus	299

G.4 The Geo System

Table G.4 Bodies in Geo System

Body	Identifier
Earth Barycenter	3
Moon	301
Earth	399

G.5 The Martian System

Table G.5 Bodies in Martian System

Body	Identifier
Mars Barycenter	4
Phobos	401
Deimos	402
Mars	499

G.6 The Jovian System

Table G.6 Bodies in Jovian System

Body	Identifier
Jupiter Barycenter	5
Io	501
Europa	502
Ganymede	503
Callisto	504
Amalthea	505
Thebe	514
Jupiter	599

G.7 The Saturnian System

Table G.7 Bodies in Saturnian System

Body	Identifier
Saturn Barycenter	6
Mimas	601
Enceladus	602
Tethys	603
Dione	604
Rhea	605
Titan	606
Hyperion	607
Iapetus	608
Phoebe	609
Helene	612
Telesto	613
Calypso	614
Methon	632
Saturn	699

G.8 The Uranian System

Table G.8 Bodies in Uranian System

Body	Identifier
Uranus Barycenter	7
Ariel	701
Umbriel	702
Titania	703
Oberon	704
Miranda	705
Uranus	799

G.9 The Neptunian System

Table G.9 Bodies in Neptunian System

Body	Identifier
Neptune Barycenter	8
Triton	801
Nereid	802
Proteus	808
Neptune	899

G.10 The Plutonian System

Table G.10 Bodies in Plutonian System

Body	Identifier
Pluto Barycenter	9
Charon	901
Nix	902
Hydra	903
Pluto	999

Appendix H

TEST RESULTS

This chapter presents the results of integrations and variational equations using the **VFSSB** and **VFIBC** performed in Chapter 4.

H.1 Integrations

H.1.1 VFSSB

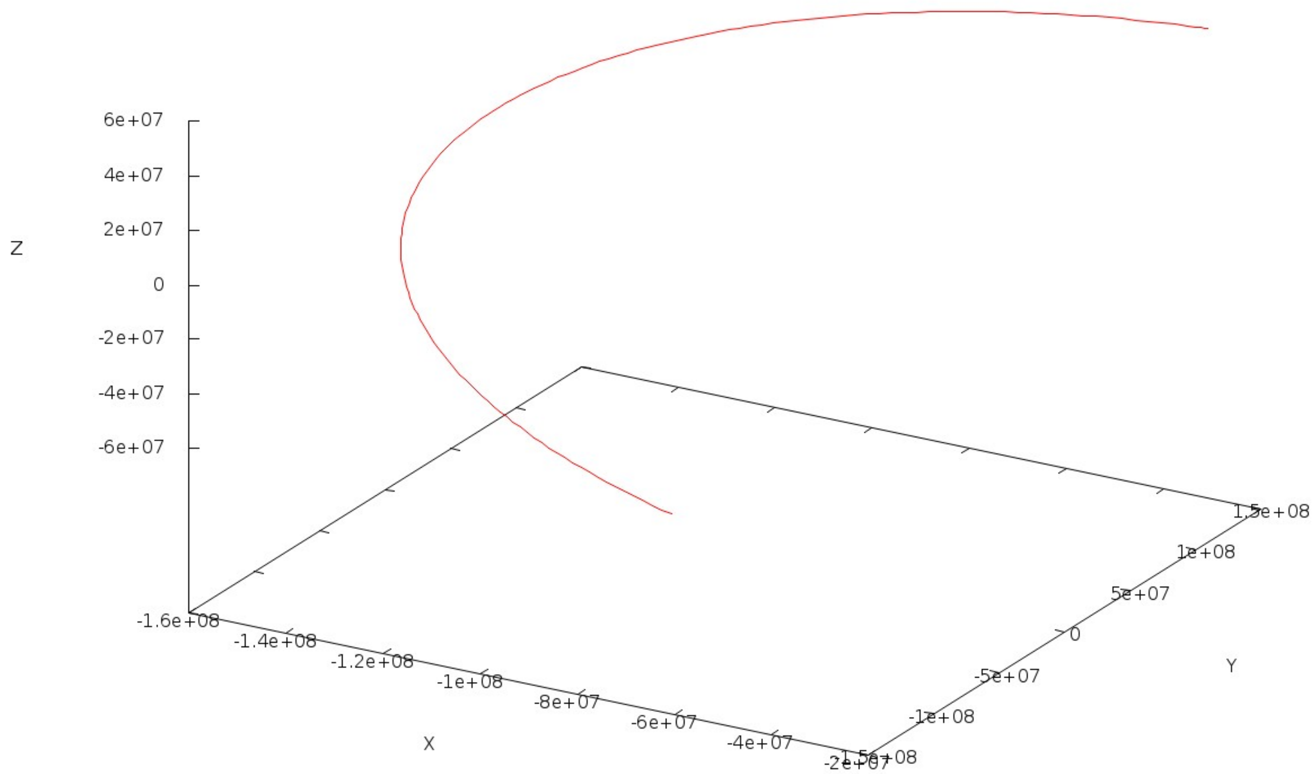


Figure H.1 VFSSB: Integrations in Equatorial Coordinates

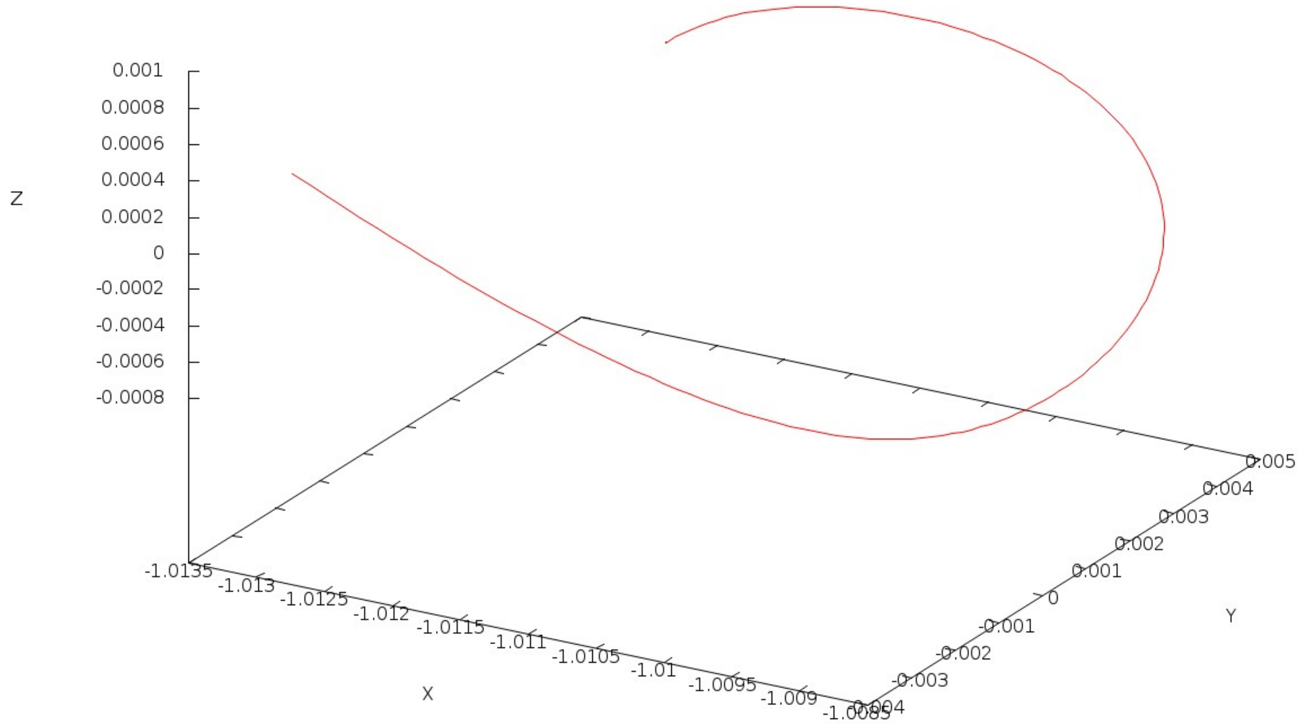


Figure H.2 VFSSB: Integrations in Adimensional Coordinates

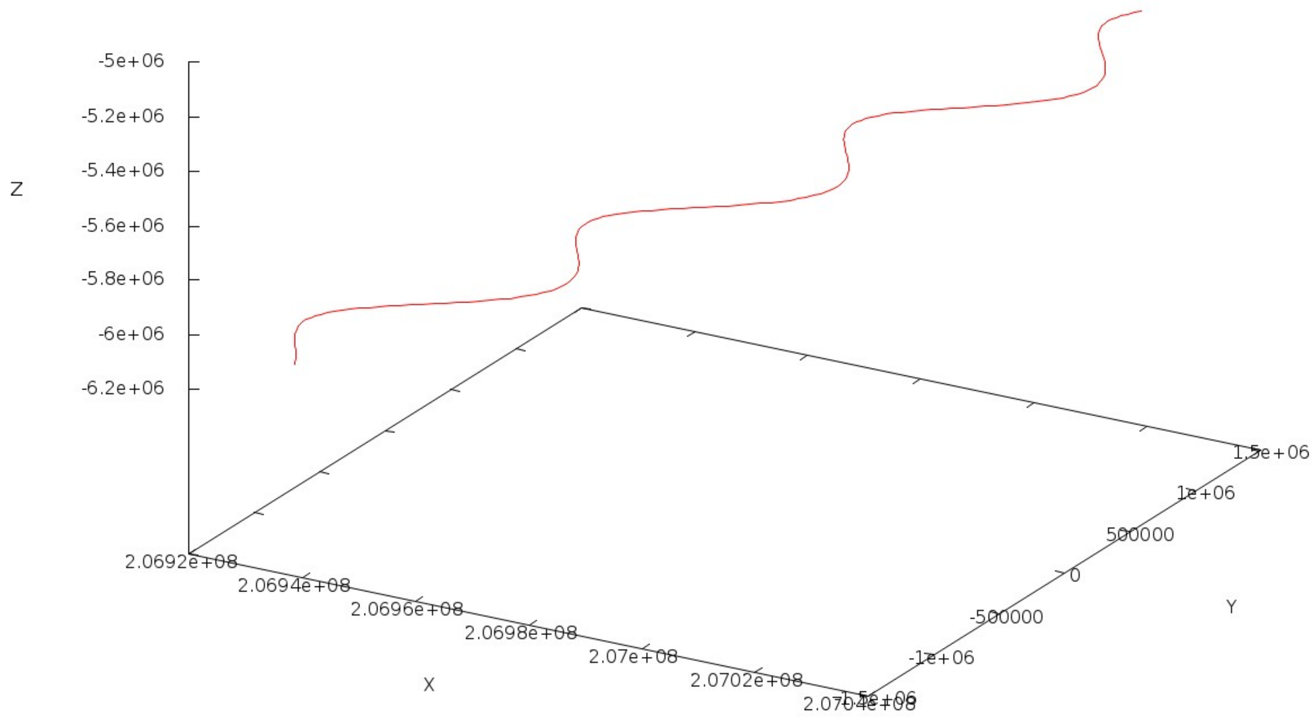


Figure H.3 Results of Integration using VFSSB for Phobos: Equatorial Coordinates

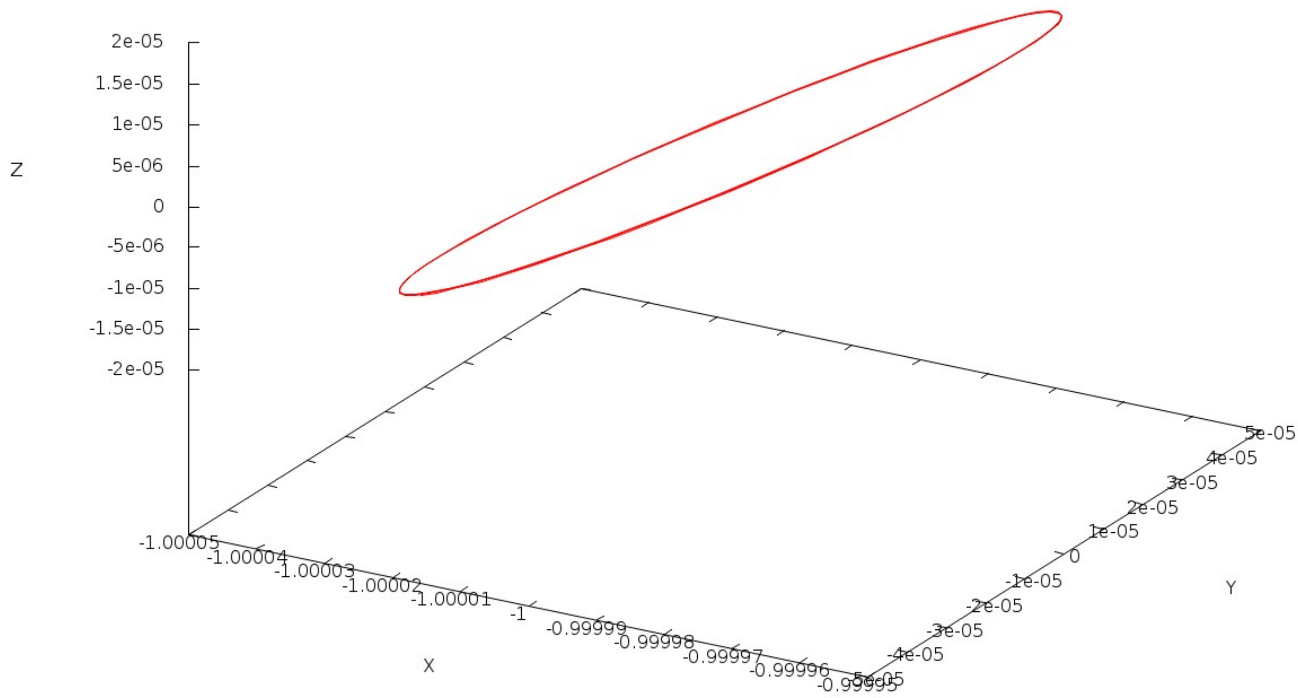


Figure H.4 Results of Integration using VFSSB for Phobos: Adimensional Coordinates

H.1.2 VFIBC

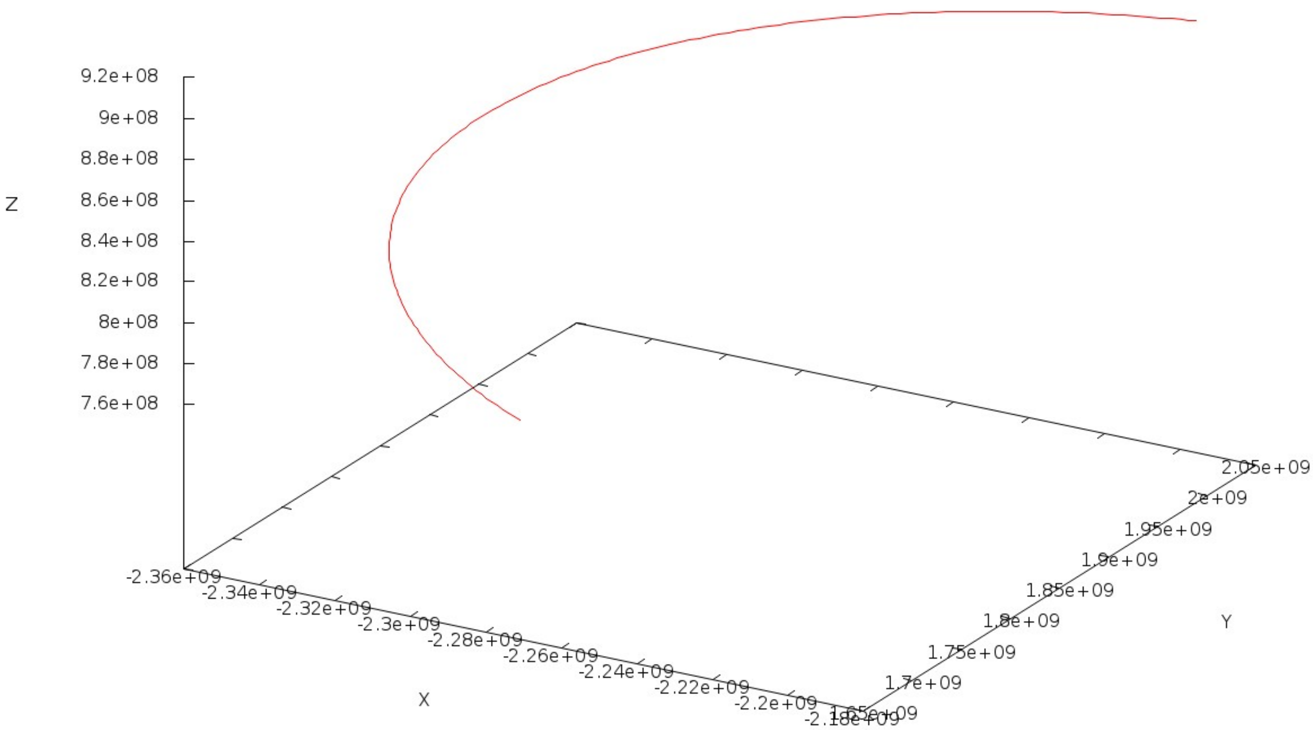


Figure H.5 VFIBC: Integrations in Equatorial Coordinates w.r.t. Uranus

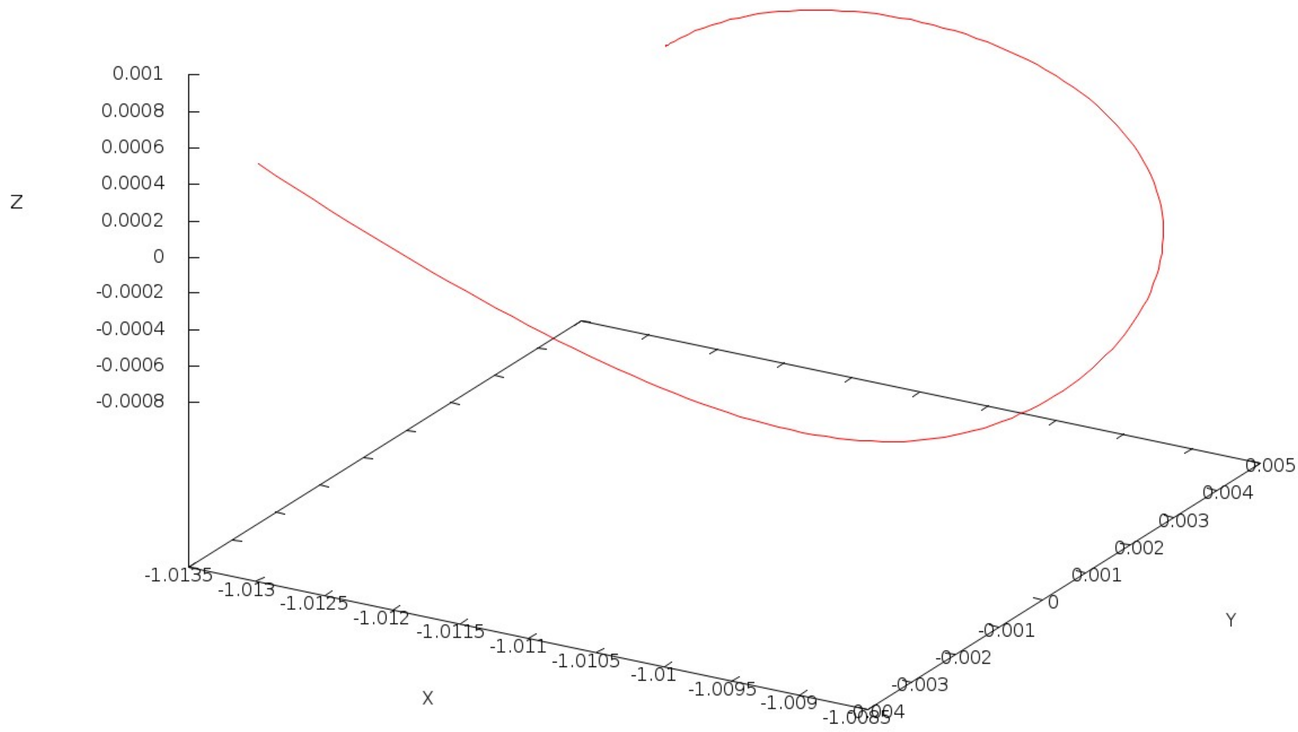


Figure H.6 VFIBC: Integrations in Adimensional Coordinates w.r.t. Uranus

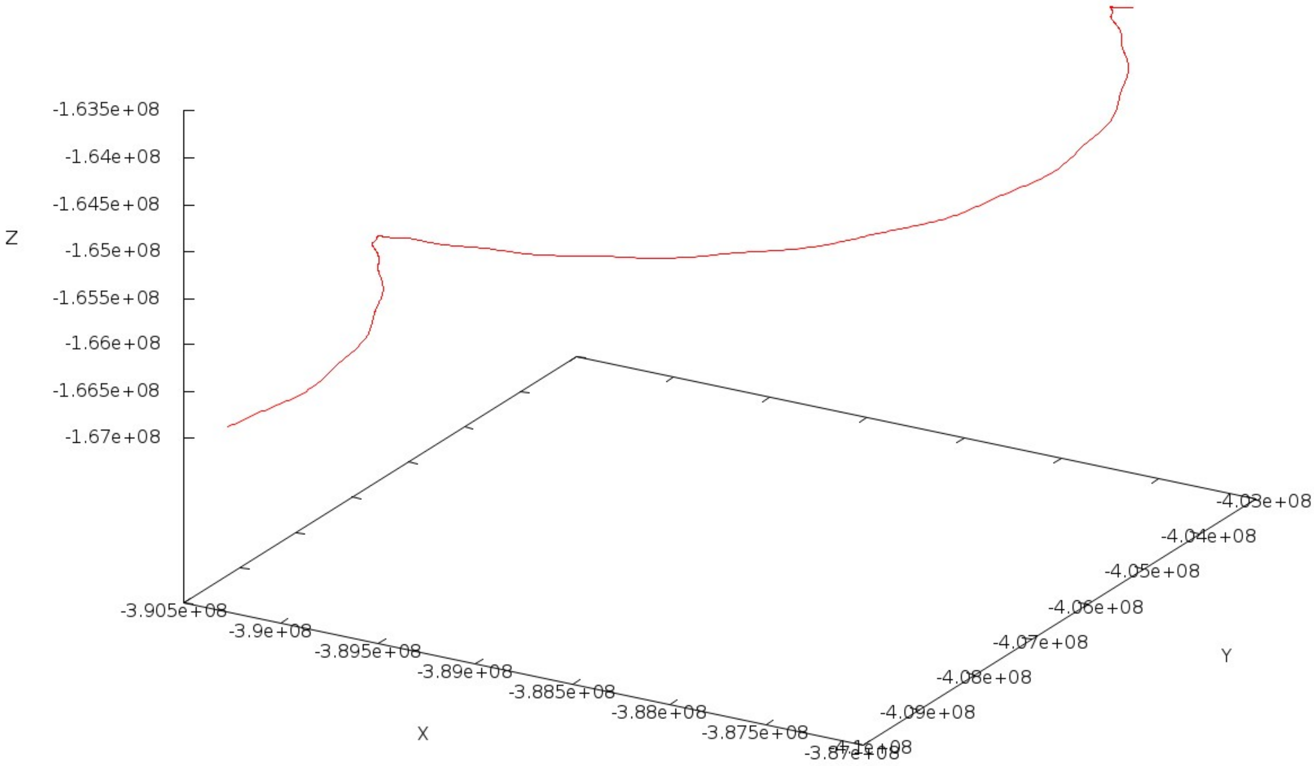


Figure H.7 VFIBC: Integrations in Equatorial Coordinates for Phobos w.r.t. Europa

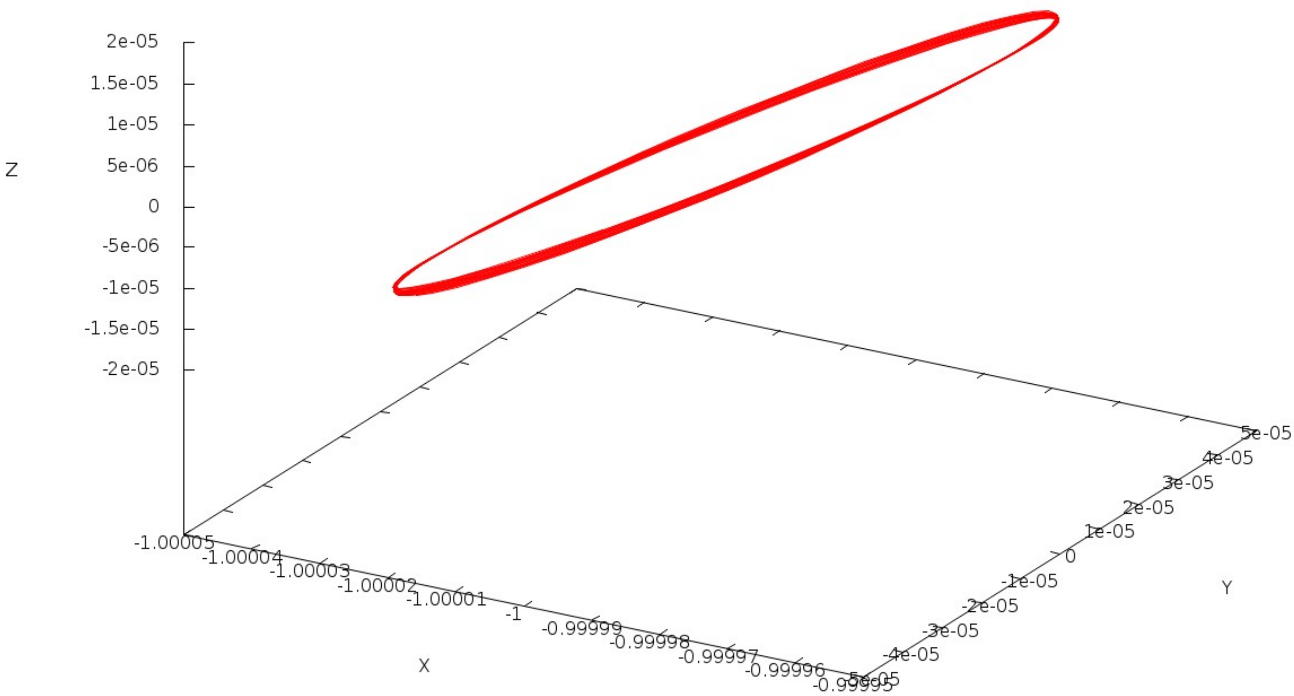


Figure H.8 VFIBC: Integrations Adimensional in Coordinates for Phobos w.r.t. Europa

H.1.3 QUASIHAMILTONIAN

Figure H.9 QUASIHAMILTONIAN: Integrations in Equatorial Coordinates

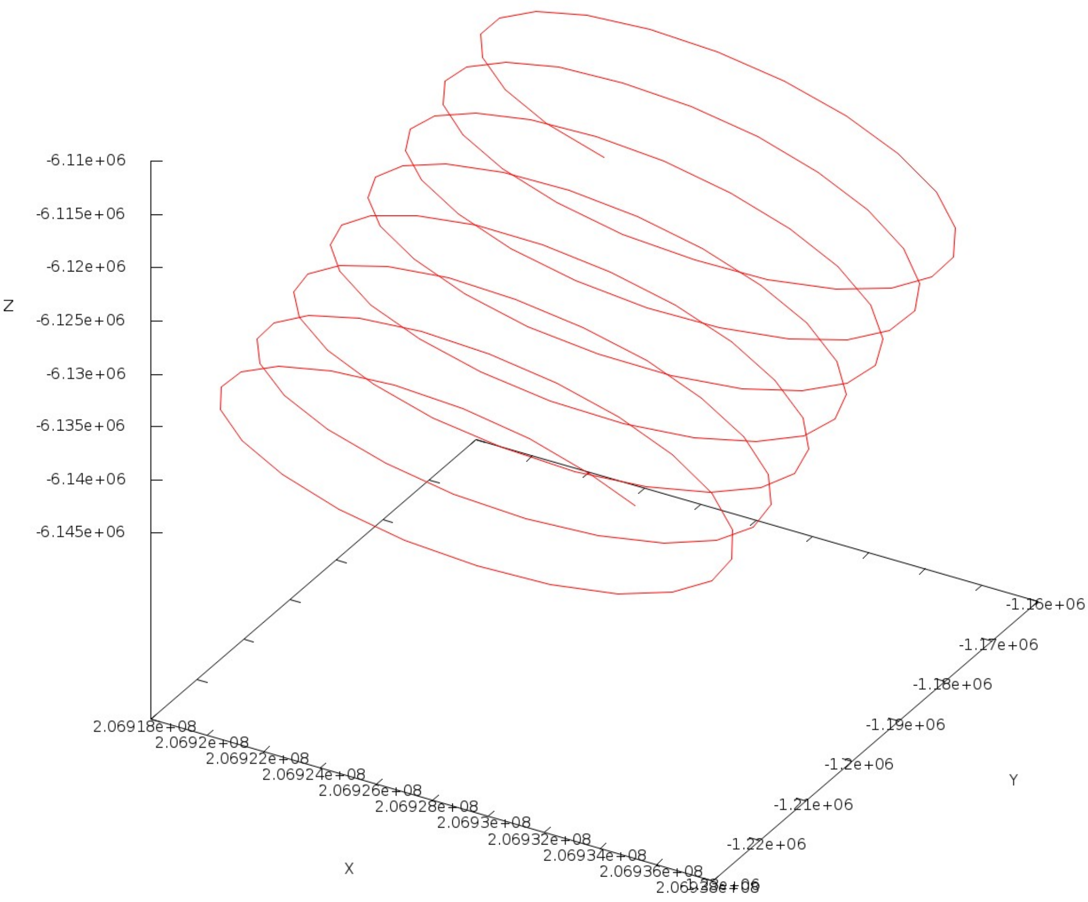
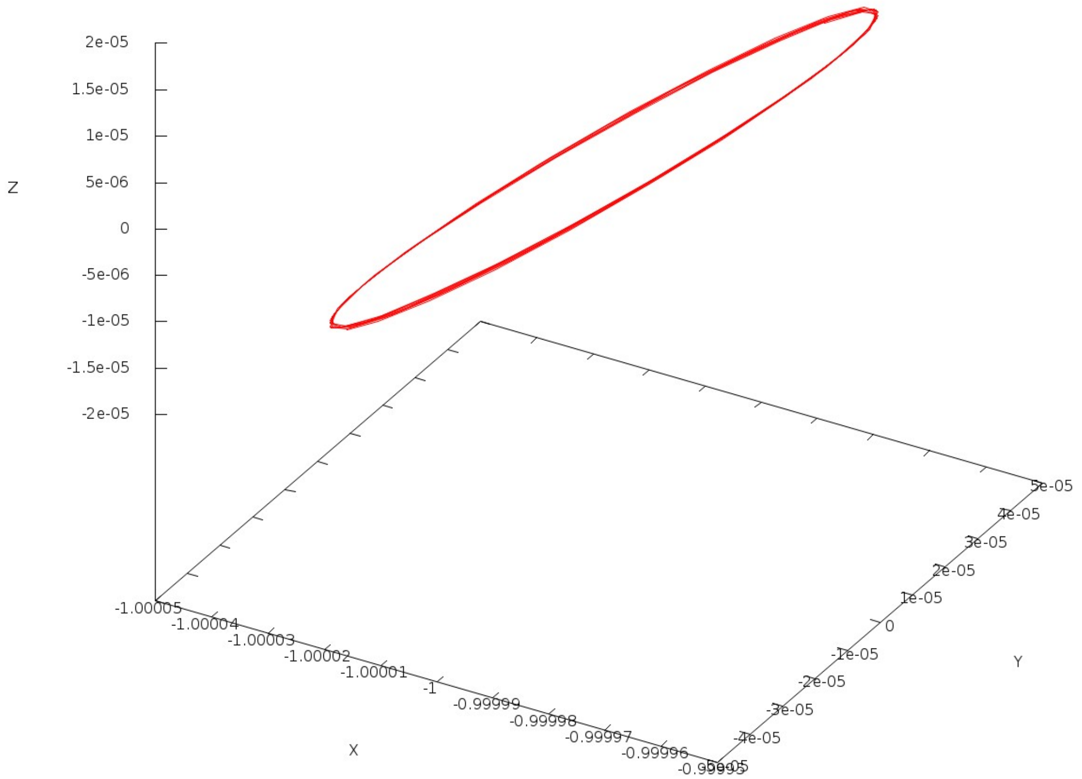


Figure H.10 QUASIHAMILTONIAN: Integrations in Adimensional Coordinates



H.2 Variational Equations

H.2.1 VFSSB

Table H.1 Maximum Relative Errors for Variational Equations from VFSSB

Initial Coordinates	Relative Error
$x = -1.01126785748736081 1$	$x = 1.55150 \times 10^{-5}$
$y = 0.00$	$y = 1.501290 \times 10^{-8}$
$z = 9.06963333171351290 0 \times 10^{-4}$	$z = 1.73978 \times 10^{-5}$
$v_x = 0.00$	$v_x = 1.37097 \times 10^{-5}$
$v_y = 9.07373420328334151 5 \times 10^{-3}$	$v_y = 1.55510 \times 10^{-5}$
$v_z = 0.00$	$v_z = 1.45060 \times 10^{-5}$
$x = -1.0000239850977075$	$x = 5.74752 \times 10^{-3}$
$y = -3.56254192881511132 \times 10^{-5}$	$y = 4.74914 \times 10^{-3}$
$z = -1.05387537084099030 \times 10^{-5}$	$z = 4.72434 \times 10^{-3}$
$v_x = 7.16956403485289884 \times 10^{-2}$	$v_x = 4.96172 \times 10^{-3}$
$v_y = -5.84134237141129761 \times 10^{-2}$	$v_y = 6.06356 \times 10^{-3}$
$v_z = 3.36939478883674440 \times 10^{-2}$	$v_z = 5.81464 \times 10^{-3}$

H.2.2 VFIBC

Table H.2 Maximum Relative Errors for Variational Equations from VFIBC

Center of Reference System	Relative Error
Venus	$x = 4.31015 \times 10^{-4}$
	$y = 1.27932 \times 10^{-3}$
	$z = 2.69458 \times 10^{-3}$
	$v_x = 2.29737 \times 10^{-4}$
	$v_y = 1.77070 \times 10^{-3}$
	$v_z = 2.35438 \times 10^{-3}$
Mercury	$x = 1.43245 \times 10^{-3}$
	$y = 1.19734 \times 10^{-3}$
	$z = 4.15500 \times 10^{-3}$
	$v_x = 1.57015 \times 10^{-4}$
	$v_y = 7.59932 \times 10^{-4}$
	$v_z = 9.30478 \times 10^{-4}$